11-29-2011

A Dynamic Approach to Pose Invariant Face Identification Using Cellular Simultaneous Recurrent Networks

Teddy Salan

A DYNAMIC APPROACH TO POSE INVARIANT FACE IDENTIFICATION USING CELLULAR SIMULTANEOUS RECURRENT NETWORKS

by

Teddy Salan

A Thesis

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

Major: Electrical and Computer Engineering

The University of Memphis

December, 2011

ACKNOWLEDGEMENTS

I would like to thank my academic advisor Dr. Khan Iftekharuddin for his dear guidance, encouragement and patience throughout my work. I want to express my gratefulness to my committee members, Dr. Peter Lau and Dr. Aaron Robinson for their supervision and assistance.

I also extend my sincere thanks to Miss. Becky Ward for her comprehensive support during my study, as well as Keith Anderson for the preparation and help he provided me, and all the members of the ISIP lab.

Last, but not least, I would like to thank all of my friends and family for their unending support and encouragement.

ABSTRACT

Salan, Teddy. M.S. Electrical and Computer Engineering. The University of Memphis. December, 2011. A Dynamic Approach to Face Identification Using Cellular Simultaneous Recurrent Networks. Khan Iftekharuddin, Ph.D.

Face recognition is a widely covered and desirable research field that produced multiple techniques and different approaches. Most of them have severe limitations with pose variations or face rotation. The immediate goal of this thesis is to deal with pose variations by implementing a face recognition system using a Cellular Simultaneous Recurrent Network (CSRN). The CSRN is a novel bio-inspired recurrent neural network that mimics reinforcement learning in the brain. The recognition task is defined as an identification problem on image sequences. The goal is to correctly match a set of unknown pose distorted probe face sequences with a set of known gallery sequences. This system comprises of a pre-processing stage for face and feature extraction and a recognition stage to perform the identification. The face detection algorithm is based on the scale-space method combined with facial structural knowledge. These steps include extraction of key landmark points and motion unit vectors that describe movement of face sequqnces. The identification process applies Eigenface and PCA and reduces each image to a pattern vector used as input for the CSRN. In the training phase the CSRN learns the temporal information contained in image sequences. In the testing phase the network predicts the output pattern and finds similarity with a test input pattern indicating a match or mismatch.

Previous applications of a CSRN system in face recognition have shown promise. The first objective of this research is to evaluate those prior implementations of CSRN-based pose invariant face recognition in video images with large scale databases. The

publicly available VidTIMIT Audio-Video face dataset provides all the sequences needed for this study. The second objective is to modify a few well know standard face recognition algorithms to handle pose invariant face recognition for appropriate benchmarking with the CSRN. The final objective is to further improve CSRN face recognition by introducing motion units which can be used to capture the direction and intensity of movement of feature points in a rotating face.

TABLE OF CONTENTS

LIST OF FIGURES

1 INTRODUCTION

Face recognition (FR) technology has been an active area of research during the last few decades and has become one of the most successful applications of image analysis. The wide range of applications of face recognition, such as biometric personal identification, man-machine communication, video surveillance and access control, and the availability of achievable technologies has sustained large-scale interest [1]. Over the years, various techniques and approaches have been developed and significant progress has been achieved in the performance of face recognition systems. Zhao et al. [1] provide one of the most comprehensive and elaborate review of FR techniques and approaches. Reference [1] also discusses advantages, limitations as well as psychophysics and neuroscience background that motivated FR development. Given the large pool of theories and techniques that are proposed for FR, it is necessary to find standard benchmarking with adequate evaluation and testing procedures for all these algorithms. Similarly, there is a need to make standardized databases publicly available for different researchers to use. For a sufficient and fair assessment, these databases contain large sets of test images with considerable variability in the data regardless of the system or application being considered. The FERET database and evaluation, described in the FERET EMFRA 2000 [2], addresses the availability of a large database of facial images and relevant methods for evaluating the performance of FR algorithms, and is a de facto standard.

There have been three FERET evaluations, with the most recent being the September 1996 FERET test. At the conclusion of the FERET program in 1997, they were succeeded by the Face Recognition Vendor Test (FRVT) experiments [3]. Together they are the most extensive evaluation of FR technology. Several educational institutions and

companies participated in these tests. Each provided and tested different algorithms using the same FERET dataset. Some of these algorithms performed very well and became baseline algorithms, such as Principal Component Analysis (PCA) [5], probabilistic eigenface with Bayesian similarity measure [6], Linear Discriminant Analysis (LDA) from UMD by Zhao et al. [7], and Elastic Bunch Graph Matching (EBGM) [8].

Both FERET and FRVT tests provided the basic models for evaluating the performance of FR algorithms in an open and a closed universe and defined the three primary face recognition tasks: *watch-list*, *identification*, and *verification*. The FRVT 2002 Performance Metrics [4] details the frameworks developed to quantify the performance of systems tested in each of the three tasks. Recognition tests are structured around sets of images. One set is the gallery G, which contains one image per every subject. An algorithm compares all images in a probe set P_g, where each person image corresponds to a person in G, and all images in an imposter set P_n, where persons have no match in G, to all the gallery images in G. This is the generalized watch-list problem. It is defined over an open universe because the system must take into account imposters from P_n not present in the watch-list and measure both the recognition and false alarm rate. The Identification and verification problems are special cases of the watch-list. Identification is a closed universe case where the set P_n is undefined, and a pure recognition rate from the set P_g specifies the performance. The verification task compares a single individual, imposter or not, with the gallery to find the match. Evaluation is based on this protocol and offers us a uniform and fair evaluation across different methods. The techniques developed in the FERET also helped define the structure of a typical FR system.

In [1] face recognition systems are divided into two groups, face recognition from still images or *static* recognition and from images sequences or *dynamic* recognition. Both systems are a complex procedure of different steps that can be summarized as segmentation (face detection/tracking), feature extraction and modeling, and face recognition (identification/verification). Algorithms that consist of all these parts are referred to as *fully automatic* algorithms [2] where the input is facial images only, and those that don't contain any preprocessing part are *partially automatic* algorithms. In the still images cases, different algorithms such as PCA and LDA are reported to have a success rate of over 95% [2].

However, these results fall short when external elements such as rotation, illumination change and occlusion are introduced. Zhao et al. [1] report that the most daunting challenges in face recognition are illumination and pose variation. In [3], FRVT 2002 results show that pose does not significantly affect performance up to $\pm 25^{\circ}$. But performance drops significantly when the pose angle reaches $\pm 40^{\circ}$. It reveals that, for the left/right rotated face, the recognition rate of the best system using Eyematic technique is only 42%. The performances of other comparable systems are less than 30%. In recent years, there have been a few face recognition techniques which attempt to address pose-invariant face recognition problems. In [8], the authors propose a face recognition method based on EBGM which assume a planar surface patch at each feature point, and learn the transformations of 'jets' under face rotation. Active shape model (ASM) [9] is a statistical flexible model obtained by learning patterns of variability from a training set of correctly annotated images. Active appearance models (AAM) [10] is another statistical model that uses both shape and gray-level information. The most unique characteristic of AAM is that it can handle even profile views in which many features are invisible. Despite the

popularity of these methods, the recognition rates for large-scale pose variations are still very low. For example, the recognition rate of EBGM with face rotated up to 90° is only 20%. In [11] the Zhou and Chellappa obtain about 60% recognition rate by driving a pose-invariant identity signature. Video-based dynamic solutions are becoming more attractive to solve this problem. The availability and temporal information contained in image sequences gives video-based FR a distinct advantage over still-image based FR [1].

In [12], temporal information of time sequences is exploited in a recurrent neural network structure for face recognition. Rather than considering just one single "snapshot" of a face, a sequence of face frames is used to perform the recognition. The temporal information encodes important information in recognition and may be very promising for addressing the large-scale pose variation problems in face recognition. Recurrent neural networks are very powerful for learning and predicting temporal information [13]. An example of such networks is a cellular simultaneous recurrent neural network (CSRN) that combines a simultaneous recurrent network (SRN) and a cellular neural network (CNN) [14]. The CSRN has been successfully exploited in solving many state transition type problems in controls applications. In [15] the CSRN demonstrates its ability to learn and predict the temporal pattern in small sequences of face images with a recognition rate of 75%. However, these results are obtained from homogeneous face sequences that contain the same pose and direction variations, and do not address complex recognition with large datasets following the standard FERET protocol.

The work done in [15] is a proof of concept for the CSRN's video-based FR capabilities on a small dataset. In this thesis, we enhance this technique with three goals. Our first aim is to scale-up the CSRN method to handle large datasets with a rigorous evaluation method following the standard FERET protocol. Next, we modify standard

image-based FR methods in literature such as PCA, LDA, Bayesian Classifier and EBGM, to deal with dynamic video sequences and compare them with our CSRN results for appropriate benchmarking. Finally, we improve our system by introducing Motion Unit vectors that identify the movement of facial feature throughout a video sequence and utilize this information in our processing.

Our proposed CSRN is a dynamic system and addresses the identification problem for video-based sequences. It is also fully automatic and includes pre-processing elements for face extraction and feature selection. In Chapter 2, we review relevant background research and the essential technologies used in this thesis. In Chapter 3, we propose our face recognition system based on CSRN by learning temporal information of face sequences. We conduct experiments using our proposed system and demonstrate improved results for large databases with varying large-pose sequences in Chapter 4. We also compare the performance of our technique with existing standard algorithms including PCA, LDA, Bayesian Classifier and EBGM. We draw our conclusions in Chapter 5.

2 Methodology

This chapter presents an extensive survey of pertinent literature and methods developed for FR. We look at the major developments, and inspect the performance and limitation of the most commonly used techniques. We present the advancements done with pose invariant recognition, and the methods applied in this thesis.

2.1 FR Techniques and Steps

FR can be seen as a challenging practice of image processing, computer vision and pattern recognition. The authors in [1] present a thorough review of face recognition technology. They express any FR system as follows: given static (still images) or dynamic (videos) representation of a scene, detect the locations where face are present and identify one or more face using a stored database of faces. Robust recognition is difficult to achieve. Faces are unique and contain so much variance in texture, shape, color, rigidity, hair, appearance, expression or makeup between different persons. In addition, when face images are captured without any constrains, outside elements such as illumination, pose variations and occlusion make recognition more difficult. In [1] the major difficulties associated with FR are summarized as follows:

1. Demographic factors: the effect of sex, age, and time-delay may affect recognition. The FERET 2002 [2] reveals that males are easier to recognize than females. Also, the greater the elapsed time between the original and new images, the less recognition is accurate.

2. Facial expression: facial expressions (e.g. happiness, sadness, disgust…) directly affect the recognition.

3. Additional artifacts or occlusion: occlusion from missing image parts, or

additional items like such as glasses, beards or any object that partially covers the face greatly increase the complexity.

4. Illumination: The same face can appear different due to a change in illumination. The changes induced by illumination are often larger than the differences between individuals [2].

5. Pose: A rotated face looses many of its key feature points, causing recognition rate to drop dramatically.

The problem of automatic face recognition involves three key steps/subtasks: (1) detection and rough normalization of faces, (2) feature extraction and accurate normalization of faces, and (3) identification and/or verification. Figure 1 shows the flow diagram of a typical face recognition system.

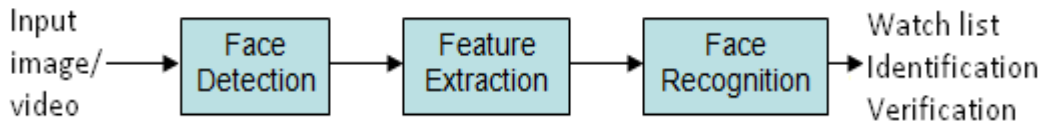

Figure 1, Configuration of a generic FR system

2.1.1 Segmentation/Face Detection

The first step is the segmentation or face detection. This problem can be defined as: Given an arbitrary image, are there any faces present in the image and if so, return the image location and extent of each face. Significant advances have been made in recent years in achieving automatic face detection under various conditions. An excellent survey of face detection and localization techniques is discussed by Yang et al. [16]. These can be put in 3 categories: template matching methods, image knowledge based method, and feature invariant approaches. Appearance or image knowledge based methods that train

machine systems on large numbers of samples have achieved the best results [17]. Knowledge-Based methods encode what humans know and see in a face and translate the human knowledge to well-defined computer rules by describing the features of a face and explaining their relationships. For example, a generic face in frontal view always contains two symmetrical eyes and ears, a nose and a mouth at almost the same relative position. These universal features of a face can be obtained by computing their corresponding distances. Jiao and Gao in [18] propose a knowledge-based method for face localization based on the location of two irises. The input frontal image is converted into binary, forming a pair of connected areas. Using knowledge about face structure, the pair of connected areas can be identified as the two eyes. With the location of two irises, the face can be extracted from the background. These methods work well with a frontal face, but with pose variations most of the key features in a face are missing. There has been much improvement in face detection for large rotation cases (more than 35°) using feature invariant approaches such as training on multiple view samples [19], or a scale-space feature extraction approach presented by Liposak and Loncaric in [20]. The scale-space method aims at locating a face in profile view with a very simple approach. The image is converted to a binary black and white image. A pre-processing step extracts the outline curve of the front of the silhouette. The Gaussian convolution of the profile line, with some particular standard deviation parameter may be used to identify and localize extrema. These extrema represent interest points like a nose peak, nose bottom, mouth point, chin point etc. Based on these key points, one can extract the face from background. In [15] Yong et al. uses a combination of scale-space and knowledge-based methods to get a pose invariant correct face extraction rate of 98%.

2.1.2 Feature Extraction

This step can be done simultaneously with the segmentation part. Most computer vision systems transform the input image data into a representative set of features. This not only allows a drastic dimensionality reduction of the input by also provides information and accurate locations of key facial features such as eyes, nose, and mouth to normalize the detected face. Zhao et al. [1] record three types of extraction methods distinguished as: (1) generic methods based on edges, lines, and curves; (2) feature-template-based methods that are used to detect facial features such as eyes; and (3) structural matching methods that take into consideration geometrical constraints on the features. Early approaches focus on localized individual features using generic methods or feature-template-based methods.

2.1.3 Recognition

Recognition is the final step in this process and is defined as a solution to either the watch-list, identification or verification problem. Zhao et al. [1] distinguish three categories of approaches used:

1. Holistic matching methods: These methods use the whole face region as the raw input to recognition systems. One example of such is the eigenface approach proposed by Turk and Pentland [5], which became the most classical and the first successful demonstration of machine face recognition. Based on principal component analysis (PCA) the algorithm extracts eigenfaces. Each distinct face in the database can be represented as a vector of weights obtained from the inner product of the image with the eigenface. When a new test image is presented, whose identification is required, we locate the face in the database whose weights are closest to the test image. The similarity is then

measured using simple Euclidian distances. PCA has two useful properties when used in face recognition. The first is that it can be used to reduce the dimensionality of the feature vectors. The second is that PCA eliminates all of the statistical covariance in the transformed feature vectors. This allows capturing the statistical properties of a face image on a global level, and makes the eigenface method more tolerant and immune to local variation and more robust than some local feature methods. Instead of the simple Euclidian distance used in [5], the standard eigenface approach extends to a Bayesian approach using a probabilistic measure of similarity as defined by the Bayesian Intrapersonal/Extrapersonal Classifier (BIC) in [6]. This algorithm aims at classifying a pair of images as *intrapersonal*, belonging to the same person, or *extrapersonal*, belonging to two different persons. In the training phase, called density estimation, the algorithm estimated the statistical properties of the two subspaces, intrapersonal and extrapersonal. When a new test image is presented, the classifier uses these density estimations as a mean of identification. Another successful holistic method is the Linear Discriminant Analysis (LDA) [7]. Using Fisher's Linear Discriminants, the basic idea of LDA is to find a linear transformation such that differences between classes are maximized and differences within classes are minimized. Using this transformation feature clusters become linearly separable.

2. Feature-based matching methods: These approaches concentrate on the geometry of local face features. One of the most successful of these systems is the Elastic Bunch Graph Matching (EBGM) [8]. EBGM first locates landmarks on an image (eyes, nose, ears). These local features are represented by wavelet coefficients called 'jets' for different scales and rotations based on fixed wavelet bases. These locally estimated wavelet coefficients are robust to illumination change, translation, distortion, rotation, and

scaling. The algorithm uses the jets are to construct face graphs, which are in turn used to find the similarity between faces.

3. Hybrid methods: Human perception doesn't simply rely on local features or global features in the whole region to recognize a face. Similarly a computer face recognition systems should use both [1]. One can argue that these methods could potentially offer the better of the two types of methods. In Pentland et al. (1994) [21], the capabilities of the earlier system [5] were extended in several directions. The concept of eigenfaces can be extended to eigenfeatures (such as eigeneyes, eigenmouth, etc) and for lower order spaces, the eigenfeatures performed better. LFA is an interesting biologically inspired feature analysis method developed by Penev and Atick (1996) [22]. LFA uses the global PCA models to extract topographic local features.

Video-based face recognition techniques are also based on these same techniques mainly the EBGM, probabilistic eigenface. Tracking is applied to improve these systems. EBGM, probabilistic eigenface and LDA methods are recognized as some of the most successful. As with other techniques mentioned above this has been proven for frontal view or near frontal view face recognition. The large-scale pose variation in face images is still a fundamental problem. In FRVT 2002 [3], the recognition rate for non-frontal images using the eigenface technique is only 42%. Neural network (NN) has been employed in face recognition.

2.2 Pose Invariant Recognition

Some solutions attempt to solve pose-invariant face recognition. These techniques can be divided into three classes [23] such as: (1) multi-view image methods, when multi-view database images of each person are available; (2) hybrid methods, when multi-view

training images are available during training but only one database image per person is available during recognition; and (3) single-image/shape-based methods where no training is carried out. Active shape model (ASM) [9] and Active appearance models (AAM) [10] are flexible statistical models obtained by learning patterns of variability from a training set of correctly annotated images. These systems are designed to handle large-rotation pose where most features are invisible or missing, therefore, different sets of features are used with respect to different pose samples at 90◦ (full profile), 45◦ (quasiprofile), and 0◦ (frontal view). When a new input face presents, all the models are used to match the image, and estimation of the pose is achieved by choosing the best fit. Despite the popularity of mentioned methods, the recognition rates of these methods for large-scale pose variations are still very low. For example, the recognition rate of EBGM with face rotated up to 90◦ is only 20% and in [11], the authors obtain about 60% recognition rate by driving a pose-invariant identity signature.

In [12], temporal information of time sequences is exploited in a recurrent neural network structure for face recognition. In this paper, rather than considering just one single “snapshot” of a face, a sequence of face is used to perform the recognition. The temporal information encodes important information in recognition and may be very promising for addressing the large-scale pose variation problems in FR. The authors in [4] also show examples for pose-invariant FR using temporal information in image sequences using a simple Elman-type recurrent neural network (RNN). They limit the pose variation of the face images between +45◦ to -45◦ for their simulation. Furthermore, they do not evaluate face recognition performance for a large scale dataset.

2.3 Review of Used Techniques

In this thesis, we introduce a dynamic face identification system to deal with large pose variations in image sequences using Cellular Simultaneous Recurrent Network (CSRN). In the pre-processing step, a scale-space feature extraction method is used to extract faces from background and locate key points. Geodesic Intensity Histogram (GIH) and Facial structure knowledge are also applied to improve this extraction. We compute the optical flow following the Horn-Schunck (HS) algorithm in image sequences to get the motion unit vectors around these key points. The Turk and Pentland eigenface technique [5] is employed to obtain pattern vectors representing each image. The combined pattern and motion vectors are fed into the CSRN. We evaluate our system following the FERET standard protocol. The subsequent sections examine the techniques used in this research.

2.3.1 Deformation Invariant GIH

In this subsection we introduce the detail about the GIH [24]. Let $I_1(x, y)$ be an image defined as I: $R^2 \rightarrow [0, 1]$. Let $I_2(u, v)$ be a deformation of I_1. This deformation is invertible, the relationship can be written as u = u(x, y), v = v(x, y), x = x(u, v), y = y(u, v), and $I_2(u, v) = I_1(x(u, v), y(u, v))$. Denote the embedding of an image $I(x, y)$ with aspect weight α as,

$$\sigma(I;\alpha) = (x' = (1-\alpha)x, y' = (1-\alpha)y, z' = \alpha I(x, y)) \tag{2.1}$$

Denote σ_1, σ_2 as the embeddings of I_1, I_2 respectively as follows,

$$\sigma_1 = (x' = (1-\alpha)x, y' = (1-\alpha)y, z' = \alpha I_1(x, y)) \tag{2.2}$$

$$\sigma 2 = (u^{'} = (1-\alpha)u, v^{'} = (1-\alpha)v, w^{'} = \alpha I_2(u,v)) \tag{2.3}$$

Let γ_1 be a regular curve on σ_1, t ∈ [a, b], and γ_2 the deformed version of this curve on σ_2 given as:

$$\gamma_1 = (x^{'}(t), y^{'}(t), z^{'}(t)) = ((1-\alpha)x(t), (1-\alpha)y(t), \alpha I\ (x(t), y(t))) \tag{2.4}$$

$$\gamma_2 = (u^{'}(t), v^{'}(t), w^{'}(t)) = ((1-\alpha)u(t), (1-\alpha)v(t), \alpha I(u(t), v(t))) \tag{2.5}$$

where,

$$w^{'} = \alpha I_2(u(t), v(t)) = \alpha I(t) = \alpha I_1(x(t), y(t)) = z(t)$$

Because the intensity is invariant to deformation, now we can study the length of γ_1, γ_2, denoted as l_1, l_2 respectively. We have,

$$l_1 = \int_b^a \sqrt{(1-\alpha)^2 x_t^2 + (1-\alpha)^2 y_t^2 + \alpha^2 I_t^2}\, dt \tag{2.6}$$

it is clear that for a large α, the curve length is dominated by the intensity changes along the curve. In the limit when $\alpha \to 1$, l_1, l_2 converge to the same value. Also, the length of curves with constant intensities tends to be trivial compared to lengths of curves with inconstant intensities. The geodesic distance, which is the distance of the shortest path between two points on the embedded surfaces, is deformation invariant. Given an interest point $p_0 = (x_0, y_0)$, the geodesic distances from it to all other points on the embedded surface $\sigma(I;\alpha)$ can be computed using the level set framework. Points with identical geodesic distances from p_0 are treated as level curves. For images defined on discrete grids, the fast marching algorithm provides an efficient method of computing these curves. Geodesic level curves provide us a way to find deformation invariant regions surrounding interest points. These regions can be used as support regions for extracting

deformation invariant descriptors. To derive invariant descriptors, we must also sample these regions using geodesic distances, to find deformation invariant sample points. In the following $\triangle$ is used to denote the sampling interval. Geodesic sampling for 2D images is done in two steps. First, the level curves are extracted at intervals of $\triangle$. Second, points are sampled from each level curve at intervals of $\triangle$. Figure 2 gives examples of 2D geodesic sampling. The interest point is the marked star in the center of the sampling. The sampled points are marked on the geodesic level curves [24].

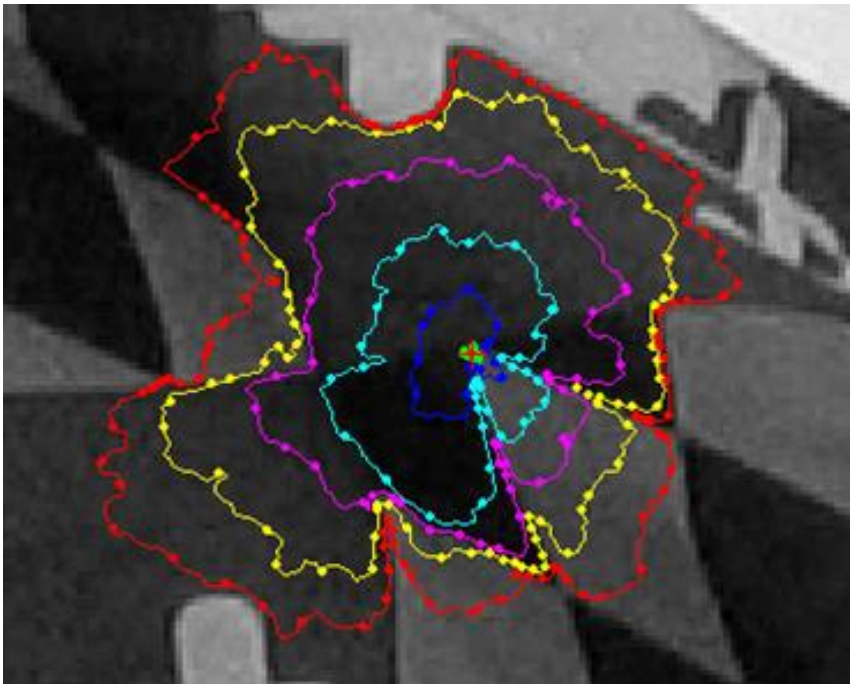

Figure 2, 2D geodesic sampling. $\alpha = 0.98$. [24]

Now we introduce the geodesic-intensity histogram (GIH), which is a deformation invariant descriptor extracted from geodesic sampling. It captures the joint distribution of the geodesic distance and the intensity of the sample points. Since both the geodesic distance and the intensity are deformation invariant, so is the GIH. Figure 3 displays examples of the geodesic-intensity histograms.

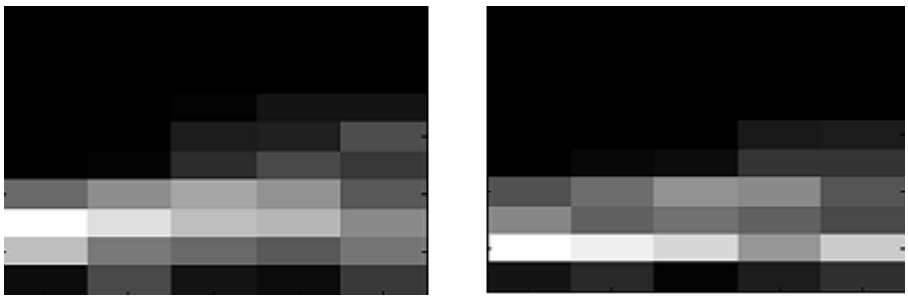

Figure 3, Geodesic-intensity histograms. [24]

Given two geodesic-intensity histogram H_p, H_q the similarity between them is measured using the χ^2 distance as,

$$\chi^2(p,q) \equiv \frac{1}{2}\sum_{k=1}^{K}\sum_{m=1}^{M}\frac{[H_p(k,m)-H_q(k,m)]^2}{H_p(k,m)+H_q(k,m)} \tag{2.7}$$

where *k* is the number of intensity intervals and *m* is the number of geodesic intervals.

2.3.2 Facial Structural Knowledge

GIH is good to calculate corresponding point in the target image, but it is a time-consuming method. Facial structure knowledge and facial texture distribution [24] can locate the key points not very accurately but more quickly. In these methods, the detecting of face is based on the location of the two irises. That is mainly because the irises are most obvious in the face image and their accuracy can be guaranteed. Firstly, the authors convert the image to binary image, and some pairs of connected areas are founded. By using the face structural knowledge, a particular pair of connected area L and R, where the two eyes are located, can be identified. The criterions are given as:

A. The L and R should be in the middle part of the image.

B. There is no connected area under the L and R in a certain distance.

C. The distance between the vertical coordinate of center points of L and R must be less than a certain value.

D. The distance between the horizontal coordinate of center points of L and R must be greater than a certain value.

Based on the location of the irises, the rough location of the other features can be obtained with the statistic knowledge of the face. Table 1 describes the distribution of facial components around two eyes, the key points such as the median point between left and right eye, the mouth center, and the nose tip can be located separately. Including two eyes, the five points are used to extract the face. Also, it is not difficult to locate more key points.

Table 1. Facial feature structure statistics [18]

Ratio to Distance of the two eyes	Eye Width	Eye to Nose Tip	Eye to Lip	Eye to Chin Tip	Nose Width	Mouth Width	Mouth Height	Eye to Eyebrow
Mean	0.42	0.51	0.88	1.59	0.57	0.70	0.27	0.35
Variance	0.03	0.05	0.04	0.09	0.05	0.06	0.02	0.05

2.3.3 Scale-space Approach for Extraction

Scale-space filtering [25] is a method that describes signals qualitatively, in terms of extrema in the signal or its derivatives, in a manner that deals effectively with the problem of scale-precisely localizing large-scale events, and effectively managing the ambiguity of descriptions at multiple scales, without introducing arbitrary thresholds or free parameters. The extrema in signal and its first few derivatives provide a useful

general-purpose qualitative description for many kinds of signals [25]. Gaussian convolution is a good description for scale issue. The Gaussian distribution is symmetric around the mean, and therefore the weights assigned to signal values decreases smoothly with distance. When a signal is convoluted with the Gaussian distribution, the signal is smoothed with respect to the scale parameter σ, the standard deviation. The signal approaches the un-smoothed signal for small σ, and approaches the signal's mean for large σ. The Gaussian is also readily differentiated and integrated.

The Gaussian convolution of signal f(x) depends both on x, the signal's independent variable, and on σ, the standard deviation. The equation is given as:

$$F(X,\sigma) = f(x) * g(x,\sigma) = \int_{+\infty}^{-\infty} f(u)\left(\frac{1}{\sigma\sqrt{2\pi}}\right) e^{-\frac{(x-u)^2}{2\sigma^2}}\, du \tag{2.8}$$

where "*" denotes convolution with respect to x. This function defines a surface on the (x, σ)-plane, where each profile of constant σ is a Gaussian-smoothed version of f(x), and the (x, σ)-plane is called scale plane [25].

The extrema in the N-th derivative of a smoothed signal can be computed by the zero-crossings in the (N+1)-th derivative. Here we use an example to explain how to construct a qualitative description of interesting points over all scales. The extrema of second derivative represent the inflection points. If we plan to locate the inflection points, we can draw the contours of the extrema at different σ. There are two simplifying assumptions to these contours such as: (1) the identity assumption, that extrema observed at different scales, but lying on a common zero-contour in scale space, arise from a single underlying event, and (2) the localization assumption, that the true location of an event giving rise to a zero-contour is the contour's x location as $\sigma \rightarrow 0$. According to these two assumptions, a coarse-scale may be used to identify an interesting point and the fine-scale

may be used to localize it. They also introduce the algorithm for nose tip location in Figure 4.

1. Extract the profile line f(x) from input image.

2. Smooth the profile line using Gaussian convolution with small parameter σ_s;

$$F(x,\sigma_s) = f(x) * g(x,\sigma_s)$$

3. Smooth the profile line using Gaussian convolution with large parameter σ_l;

$$F(x,\sigma_l) = f(x) * g(x,\sigma_l)$$

4. Flatten the profile line; $ff(x,\sigma_s,\sigma_l) = F(x,\sigma_s) - F(x,\sigma_l)$

5. Compute Gaussian convolution with σ_n;

$$FN(x,\sigma_n,\sigma_s,\sigma_l) = ff(x,\sigma_s,\sigma_l) * g(x,\sigma_n)$$

6. Locate extrema in $FN(x,\sigma_n,\sigma_s,\sigma_l)$.

7. Use criteria $((a_1 > 0)$ & $(a_2 < 0)$ & $(b_1 > b_2))$ to locate nose tip. The criteria is shown in Figure 5.

Figure 4, Algorithm of the scale-space method to locate nose tip [20].

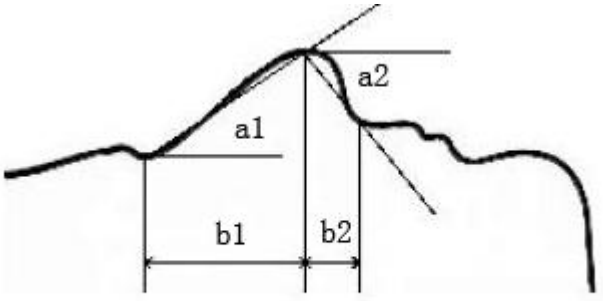

Figure 5, Criteria for Nose tip [20]

2.3.4 Optical Flow

Optical flow is the distribution of movement velocities of brightness patterns across an image [29]. It can arise from relative motion of objects and viewer, so it could give important information about the spatial arrangement of the objects viewed and the rate of change of this arrangement.

Optical flow cannot be computed locally, and independently from the neighborhood, some assumed constraints would be given when the algorithms are explained. In the optical flow field, the Horn-Schunck (HS) algorithm, detailed in [29], is a classical and easy to implement method.

Denote $E(x,y,t)$ represents the image brightness at the point (x,y) at time t. The brightness of a particular point in the pattern is constant, such that

$$\frac{dE}{dt} = 0 \tag{2.9}$$

and,

$$\frac{\delta E}{\delta x}\frac{dx}{dt} + \frac{\delta E}{\delta y}\frac{dy}{dt} + \frac{\delta E}{\delta t} = 0 \tag{2.10}$$

We will let,

$$u = \frac{dx}{dt} \text{ and } v = \frac{dy}{dt} \tag{2.11}$$

Let's define a signal linear equation which has two unknown parameters u and v.

$$E_x u + E_y v + E_t = 0 \tag{2.12}$$

The HS algorithm assumes smoothness in the distribution of brightness velocities [29].

This constraint can be expressed by limiting the difference between the flow velocity at a point and the average velocity over a small neighborhood containing the point. Equivalently we can minimize the sum of the squares of the Laplacians of the x and y components of the flow. The Laplacians of u and v:

$$\nabla^2 u = \frac{\delta^2 u}{\delta x^2} + \frac{\delta^2 u}{\delta y^2} \text{ and } \nabla^2 v = \frac{\delta^2 v}{\delta x^2} + \frac{\delta^2 v}{\delta y^2} \tag{2.13}$$

In simple situations, both Laplacians are zero. We must estimate the derivatives of brightness from the discrete set of image brightness measurements available. HS uses a set, which gives estimate of E_x, E_y, E_t at a point in the center of a cube formed by eight measurements. Each of the estimates is the average of four first differences taken over adjacent measurements in the cube [29] as follows,

$$\begin{aligned}
E_x &\approx \frac{1}{4}\{E_{i,j+1,k} - E_{i,j,k} + E_{i+1,j+1,k} - E_{i+1,j,k} + \\
&E_{i,j+1,k+1} - E_{i,j,k+1} + E_{i+1,j+1,k+1} - E_{i+1,j,k+1}\} \\
E_y &\approx \frac{1}{4}\{E_{i,j+1,k} - E_{i,j,k} + E_{i+1,j+1,k} - E_{i+1,j,k} + \\
&E_{i,j+1,k+1} - E_{i,j,k+1} + E_{i+1,j+1,k+1} - E_{i+1,j,k+1}\} \\
E_t &\approx \frac{1}{4}\{E_{i,j+1,k} - E_{i,j,k} + E_{i+1,j+1,k} - E_{i+1,j,k} + \\
&E_{i,j+1,k+1} - E_{i,j,k+1} + E_{i+1,j+1,k+1} - E_{i+1,j,k+1}\}
\end{aligned} \tag{2.14}$$

The unit of the length in the reference is the grid spacing interval in each image frame, and the unit of time is the image sampling period as shown in Figure 6.

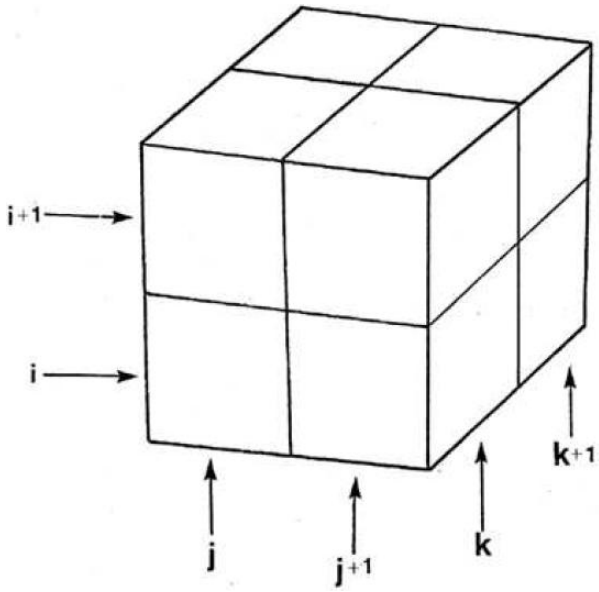

Figure 6, The three partial derivatives of image brightness at the center of the cube are each estimated from the average of first differences along four parallel edges of the cube. Column index *j* corresponds to the *x* direction in the image, row index *i* to the *y* direction, while *k* lies in the time direction. [29]

We also approximate the Laplacians of *u* and *v* given as,

$$\nabla^2 u \approx k(\bar{u}_{i,j,k} - u_{i,j,k}) \text{ and } \nabla^2 v \approx k(\bar{v}_{i,j,k} - v_{i,j,k}) \tag{2.15}$$

where variable *k* is the proportionality factor and equals 3, and the local averages $\bar{u}$ and $\bar{v}$ are defined as follows (Figure 30):

$$\begin{aligned}
\bar{u}_{i,j,k} &= \frac{1}{6}\{u_{i-1,j,k} + u_{i,j+1,k} + u_{i+1,j,k} + u_{i,j-1,k}\} \\
&+ \frac{1}{12}\{u_{i-1,j-1,k} + u_{i-1,j+1,k} + u_{i+1,j+1,k} + u_{i+1,j-1,k}\} \\
\bar{v}_{i,j,k} &= \frac{1}{6}\{v_{i-1,j,k} + v_{i,j+1,k} + v_{i+1,j,k} + v_{i,j-1,k}\} \\
&+ \frac{1}{12}\{v_{i-1,j-1,k} + v_{i-1,j+1,k} + v_{i+1,j+1,k} + v_{i+1,j-1,k}\}
\end{aligned} \tag{2.16}$$

The sum of the total errors is minimized

$$\varepsilon^2 = a^2 \varepsilon_c{}^2 + \varepsilon_b{}^2 \tag{2.17}$$

The a^2 is a weighting factor. Image brightness measurements may be corrupted by quantization error and noise so we cannot expect ε_b to be identically zero. This quantity will tend to have an error magnitude that is proportional to the noise in the measurement. That is why a^2 is chosen in Eq. (2-17). The other terms are given as,

$$\varepsilon_b = E_x u + E_y v + E_t \tag{2.18}$$

and,

$$\varepsilon_c{}^2 = (\vec{u} - u)^2 + (\vec{v} - v)^2 \tag{2.19}$$

with iterative solution as follows,

$$u^{n+1} = \bar{u}^n - \frac{E_x[E_x \bar{u}^n + E_y \bar{v}^n + E_t]}{(a^2 + E_x{}^2 + E_y{}^2)} \tag{2.20}$$

and,

$$v^{n+1} = \bar{v}^n - \frac{E_y[E_y \bar{u}^n + E_x \bar{v}^n + E_t]}{(a^2 + E_x{}^2 + E_y{}^2)} \tag{2.21}$$

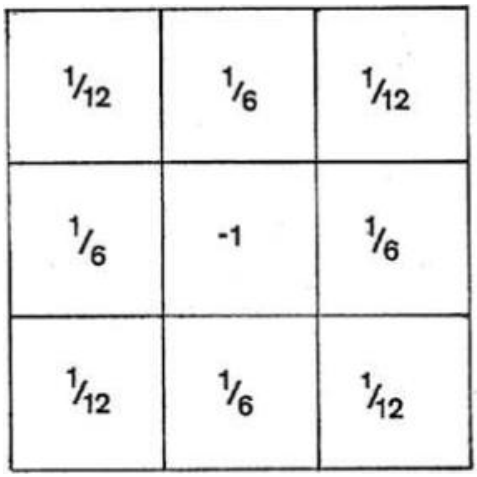

Figure 7, The Laplacian is estimated by subtracting the value at a point from a weighted average of the neighboring points [29]

2.3.5 Cellular Simultaneous Recurrent Network

CSRN is a hybrid of a SRN and a cellular neural network. This combination offers a powerful network that is able to solve the maze navigation problem as shown in [14]. To fully understand the CSRN we must first take a look at its two major parts. The SRN part of the CSRN, by itself, has been proven to be more powerful then multi layer perceptions (MLPs). Previous research has shown that functions generated by MLPs are always able to be learned by SRNs, but the opposite is not true. The recurrent nature of the SRN is similar to the way the brain works. SRNs use the output of the current iteration as input for the next iteration. This makes them excellent for prediction problems since they are able to learn from previous iterations. The basic topology of the SRN can be seen in Figure 8. The network is a function of W, x and z where W are the weights x is the external input and z is the output feedback.

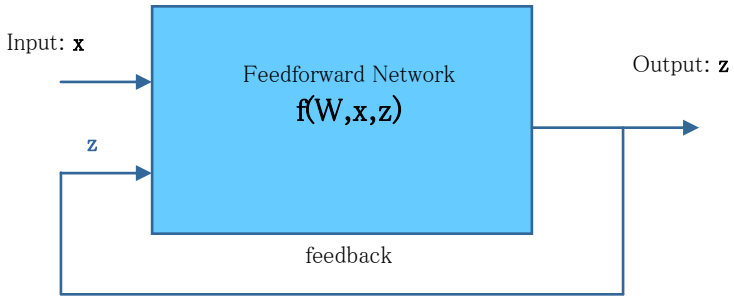

Figure 8, The basic topology of SRN

Cellular neural networks (CNN) consist of identical elements, arranged in some sort of geometry (Figure 9). This configuration reduces the required number of weights. Due to the symmetry of the network, each element is able to share the same weights. Decreasing the number of weights can significantly decrease the time needed to train the network. The symmetry of cellular neural networks can also be useful in solving problems that contain a similar sort of inherent geometry. Each element of such a network can be as simple as an artificial neuron or more complex, as a MLP.

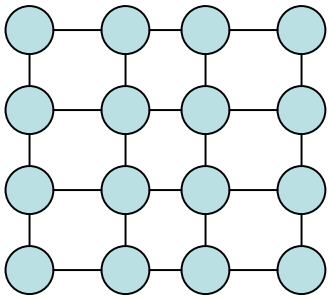

Figure 9, A typical cellular architecture

These two networks, SRN and CNN, create the CSRN. The idea of CSRN is biologically motivated. The behavior of CSRN mimics the cortex of the brain which consists of columns similar to each other. The CSRN is, at one time, trained with back

propagation through time (BPTT). However, BPTT is very slow. In [14] the extended Kalman filter (EKF) is implemented to train the network by state estimation. The architecture of the CSRN is shown in Figure 10.

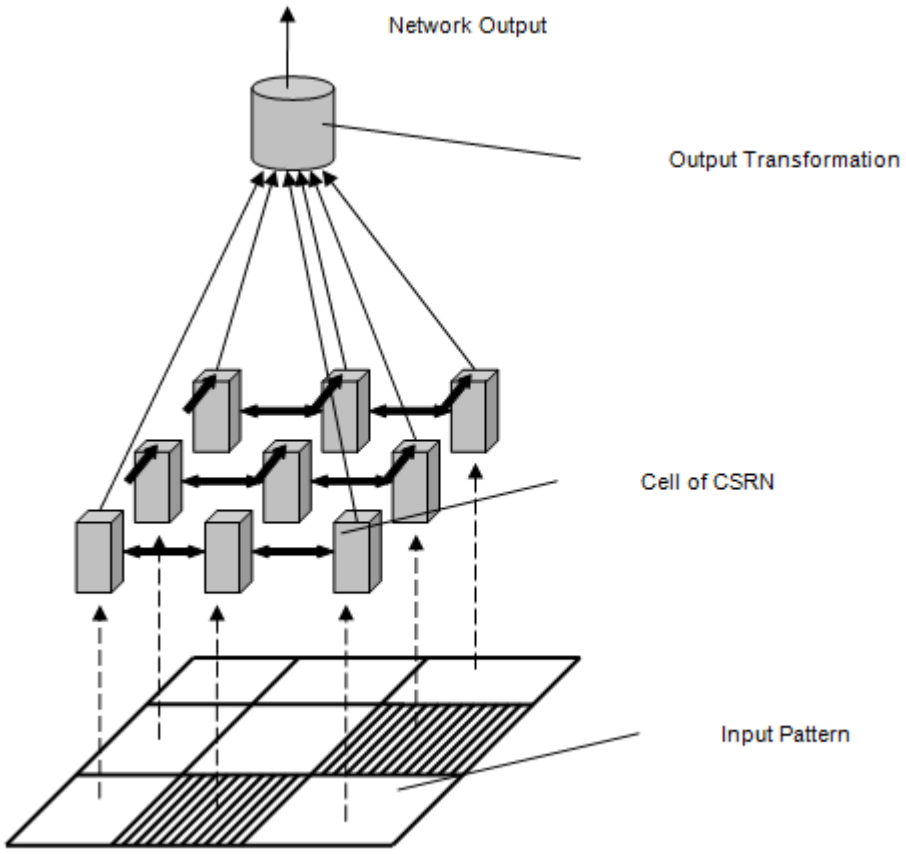

Figure 10, CSRN architecture

2.3.6 Eigenface

In general, FR techniques can be categorized into three major classes such as: feature based matching, holistic based and geometric methods. Feature based recognition uses the position, size and relationship of facial features (eyes, nose, and mouth) to perform face recognition. The holistic approach recognizes faces using the global 2-D patterns of the face. Some of them also use 3-D models of the face image for recognition. Most of these techniques are computationally intensive and not very accurate for large

scale pose variation in facial images. Eigenface for recognition seeks to implement a system capable of efficient, simple, and accurate FR in a constrained environment. The system does not depend on 3-D models or intuitive knowledge of the structure of the face. Classification is performed using a linear combination of characteristic features, known as the eigenface.

In general, let's consider the size of a face image is N, where N = m × n and (m, n) are the width and height of this image. Let's also consider a set of M images ($\Gamma_1, \Gamma_2, \ldots, \Gamma_M$), each image in this set has the unified size N. The mean of the images is the "average face" given as,

$$\psi = \frac{1}{M}\sum_{i=1}^{M}\Gamma_i \tag{2.22}$$

Each training image differs from the average face by,

$$\Phi_i = \Gamma_i - \psi \tag{2.23}$$

The eigenvectors and eigenvalues of the new face space Φ can be calculated from covariance matrix of the face images as,

$$C = \frac{1}{M}\sum_{i=1}^{M}\Phi_i\Phi_i^T = AA^T \tag{2.24}$$

where A = [$\Phi_1\ \Phi_2 \ldots \Phi_M$] and C are N × N matrices. Computing the N eigenvectors of C is computationally expensive. However, because the number of the face images is much smaller than the dimension of the face image (M << N), there are only M-1 nontrivial eigenvectors. So if we consider a smaller matrix L= $A^T A$, it will yield a smaller number of eigenvectors for M. $A^T A\, V_i = \lambda_i V_i$,, where λ_i is the eigenvalue and V_i is the eigenvectors of $A^T A$, for $A(A^T A)\, V_i = \lambda_i\, (AV_i)$, and $(AA^T)(A\, V_i) = \lambda_i\, (AV_i)$, therefore,

(AV_i) is the eigenvector of C and also we can call it eigenface. Hence, the eigenface is given as,

$$\mathrm{U}_i = (AV_i) = [\,\Phi_1\ \Phi_2 \ldots \Phi_M\,] \begin{bmatrix} \upsilon_1^i \\ \upsilon_2^i \\ . \\ . \\ \upsilon_M^i \end{bmatrix} = \sum_{k=1}^{M} \upsilon_k^i \Phi_k \tag{2.25}$$

where (i=1,…, M'), $M' \le M - 1$

The set of eigenface U = $[U_1 U_2 ... U_{M'}]$ is the eigenface subspace. The new image Γ is projected into the eigenface subspace using a weight function given as,

$$\omega_k = \frac{U_k^T (\Gamma - \Psi)}{\lambda_k} \qquad \text{k=1,…, } M' \tag{2.26}$$

The weights form a vector $\Omega = [\omega_1 \omega_2 ... \omega_{m'}]$, and it is known as the pattern vector of Γ. The face can be represented by its pattern vector and its M' corresponding eigenfaces. The Euclidian distance between the new image and a class of faces k is $\varepsilon_k = \|\Omega - \Omega_k\|$. If the distance ε_k is less then than a threshold θ, the face is assigned to recognized, and assigned to class k. This threshold is assigned empirically. One of the drawbacks of eigenface method is that this method is mostly limited to the images that taken from a very similar viewpoint, commonly, the frontal view. This substantially limits a model's robustness and effectiveness [3]. In our work, we attempt to address this limitation of eigenface method using CSRN for large facial pose variations in image sequences.

2.3.7 Temporal Information

In a set of image sequences illustrated in Figure 11, all the sequences start from and stop at two particular ends, and each identical frame in every sequence has the nearly same rotation angle, then we can divide the face space into a set of subspaces associated with time index. As shown in Figure, we have 5 subspaces, and each frame contributes to one subspace. The difference between successive frames can be calculated as "Temporal Signature" [12].

If we use a pattern vector Ω to represent an image frame, the Euclidian distance between two successive frames is given as: $\varepsilon = \left\| \Omega_t - \Omega_{t+1} \right\|$ where t is the frame index. We consider this distance ε as the temporal signature in a face class.

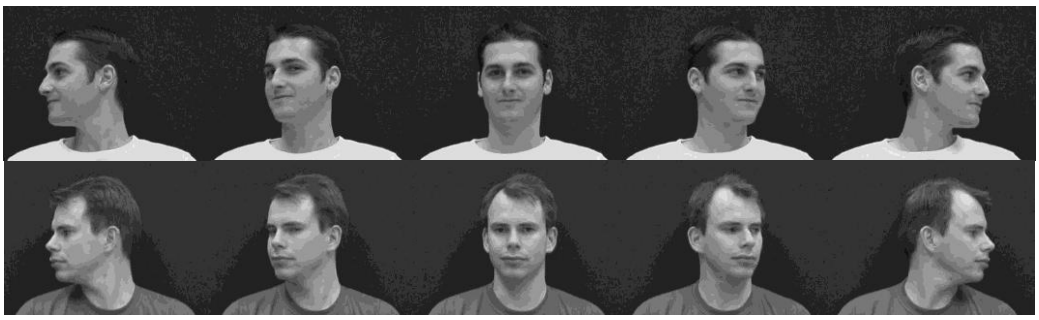

Figure 11, Images sequence.

The two image sequences in Figure 11 are taken from two subjects, each contains 5 frames, rotates from right to left. Each column presents a frame.

2.4 Databases

The FERET program set out to establish a large database of facial images that was gathered independently from the algorithm developers [2]. The images were collected in a

semi-controlled environment. To maintain the consistency throughout the database, the same physical setup was employed in each photography session. The FERET database was collected in 15 sessions between August 1993 and July 1996. The database contains 1564 sets of images for a total of 14,126 images that includes 1199 individuals and 365 duplicate sets of images [2]. 200 individuals in this database contain multi-degree faces. The degrees change from left profile to right profile at $90^0, 60^0, 40^0, 25^0, 15^0$, and 0^0. However these are still images of rotated faces. The database does not contain large sequences of a face rotating. For this research we used image sequences from the VidTIMIT [26] database. It is composed of video and corresponding audio recordings of 43 people, reciting short sentences. It can be useful for research on topics such as multi-view face recognition, automatic lip reading and multi-modal speech recognition. Each person performs a head rotation sequence in each session. The sequence consists of the person moving their head to the left, right, back to the center, up, then down and finally return to center. Figure 12 shows the example of VidTIMIT database.

Figure 12, Example of VidTIMIT database sequences

2.5 Evaluation

The evaluation protocol of FRVT 2002 [27] became the standard framework used for quantitative evaluating the performance of FR systems and techniques. It defines how different systems and datasets should be used for separate recognition tasks. Some

conceptual datasets G, P_g, and P_n are employed. G is a gallery set that contains one unique signature (image) per subject and represents the set of images that have been enrolled in a recognition system. P_g is a probe set, each of its images has a match in the gallery, and represents a legitimate user. The images in the imposter set P_n, do not have a match in the gallery. This set represents persons intending to deceive the system. A match describes the comparison of probe and gallery images of the same individual. A non-match means images coming from different persons. Grother et al. [27] use the similarity score to measure the similarity between querry image and target image. A face recognition algorithm must construct a similarity matrix based on the similarity score, whose ijth element is the similarity between the ith element of G and the jth element of P_g or P_n. The matrix is stored in column order; each column corresponds to an unknown query image being compared with all the target images. All the results of FRVT 2002 are obtained from the similarity elements corresponding to the rows defined by the subset G, and the columns defined by P_g and P_n [27].

In FRVT 2002, the authors design three relevant tasks to simulate different application such as identification, verification, and watch list. They also present separate appropriate statistics for each. As described above, performance on each of these tasks is obtained solely from the similarity values extracted from the similarity matrix and from the subject identities [27]. The watch list problem is a generalization of both identification and verification. In this task, a probe (consists of P_g and P_n) is compared to a gallery. The system identifies each probe image, and rank them based on the similarity score. However, a system could not identify the individuals who are not on the watch list. Therefore, we should measure a false alarm rate. This makes clear that the generalized

watch list problem is defined over an open-universe. In the following, we will detail the three tasks: identification, verification, and watch list.

1. Watch list: Conducted by using a watch list G and two probe sets: P_g with subjects who are all legitimate and, P_n with subjects who are all impostors. The identification rate equal as the fraction of probes in P_g that are detected at or above threshold t and recognized at rank r or better as follows,

$$P_{id}(t,r) = \frac{\left|\{p_j : rank(p_j) \le r, S_{ij} \ge t, id(p_j) = id(g_j)\}\right|}{\left|Pg\right|} \quad \forall p_j \in P_g \tag{2.27}$$

Here S_{ij} is the similarity of the target and the probe and the rank is defined as the number of watch list entries which have greater than or equal similarity to the probe than the matching entry. The rank is given as,

$$rank(p_j) = \left|\{g_k : s_{kj} \ge s_{ij}, id(g_i) = id(p_j)\}\right| \quad \forall g_k \in G \tag{2.28}$$

The False alarm is computed as the fraction of probes from P_n whose similarity to any gallery image is at or greater than threshold. The False alarm is given as,

$$P_{FA}(t) = \frac{\left|\{p_j : \max_i S_{ij} \ge t\}\right|}{\left|Pn\right|} \quad \forall p_j \in P_n \quad \forall g_i \in G \tag{2.29}$$

2. Identification: In this task the probes only come from P_g, therefore, we can use identification rate to mark the performance of system. The identification rate is then stated as the fraction of probes whose rank is at r or lower:

$$P_I(r) = \frac{\left|C(r)\right|}{\left|P_g\right|} \tag{2.30}$$

where C(r) is,

$$C(r) = \left|\{p_j : rank(p_j) \le r\}\right| \quad \forall p_j \in p_g \tag{2.31}$$

3. Verification: a single probe image is compared with a gallery image and the similarity score is compared against a threshold to verify the individual or otherwise. Two types of error may occur in this process such as firstly, a false reject in which the system incorrectly rejects the individual below threshold; and secondly, a false accept in which an imposter is accepted by the system above threshold.

$$P_V(t) = \frac{\left|\{p_j : S_{ij} \ge t, id(p_j) = id(g_j)\}\right|}{|Pg|} \quad \forall p_j \in P_g \tag{2.32}$$

The Receiver Operating Characteristic (ROC) is computed to quantify verification performance [25]. It shows the tradeoff between the two types of error by plotting estimates of the verification rate, P_V against the false accept rate P_{FA}. The verification rate is the fraction of probes whose similarity scores are greater than or equal to the threshold value, which is given as,

$$P_{FA}(t) = \frac{\left|\{S_{ij} : S_{ij} \ge t\}\right|}{|Pn||G|} \quad \forall p_j \in P_n \quad . \tag{2.33}$$

3 Proposed Methods

In this chapter, we present our proposed methods to dynamic face identification with large-scale pose variations using a CSRN. The system is fully automatic and operates on input image sequences of rotating faces. As discussed in Chapter 1, we innovate on the pose invariant FR in Ref. [15]. The segmentation step extracts and localizes the face in an image. The scale-space method and facial structure knowledge are used to extract faces of different pose variations. The GIH is exploited to improve the accuracy of extraction. With the face localization step we perform a feature extraction to locate key points in a face and calculate the motion units at these points. We use eigenface and PCA to obtain reduced pattern vectors representing each image. Combined with the motion units, these vectors are the input to CSRN. The CSRN is used to learn and capture temporal information in image sequences, then predict the testing sequence. The identification is based on the similarity of these sequences with the input sequences based on Euclidian distance. This procedure is summarized in Figure 13.

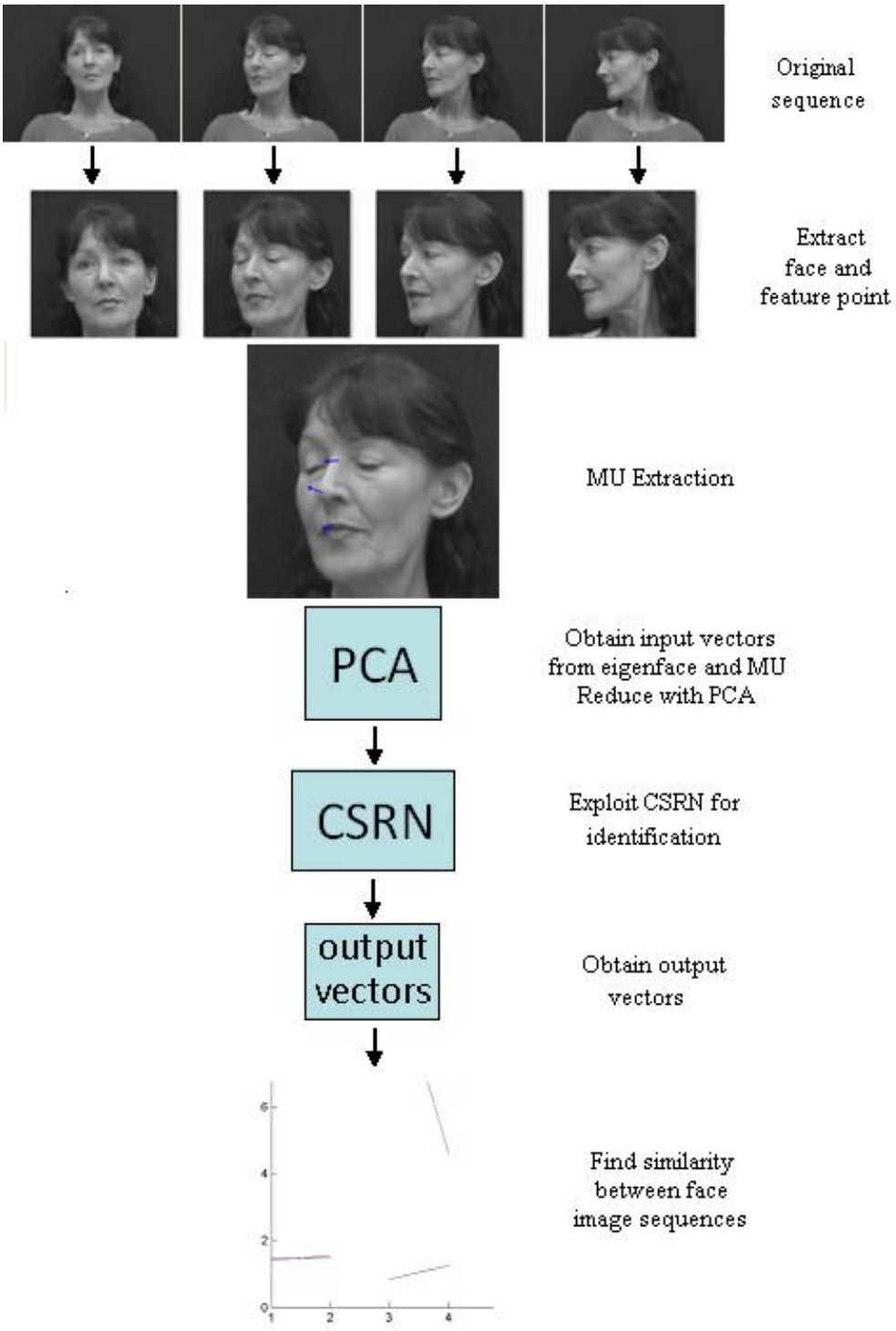

Figure 13, Overall flow diagram of proposed face recognition system based on CSRN.

The identification rate is compared with standard algorithms provided by the Colorado State University (CSU) face identification toolbox [28], with slight modifications to adjust for dynamic identification.

3.1 Preprocessing

Our aim is to have a pose invariant face and feature extraction system. We extract faces following a combination of the scale-space method and the knowledge-based key point extraction coupled with GIH as proposed in [15]. Figure 14 shows the flow chart of our extraction system. In addition we use these methods to extract the locations of key features and compute the optical flow around them, obtaining motion unit vectors. These will be essential to determine the direction and intensity of motion in the sequence.

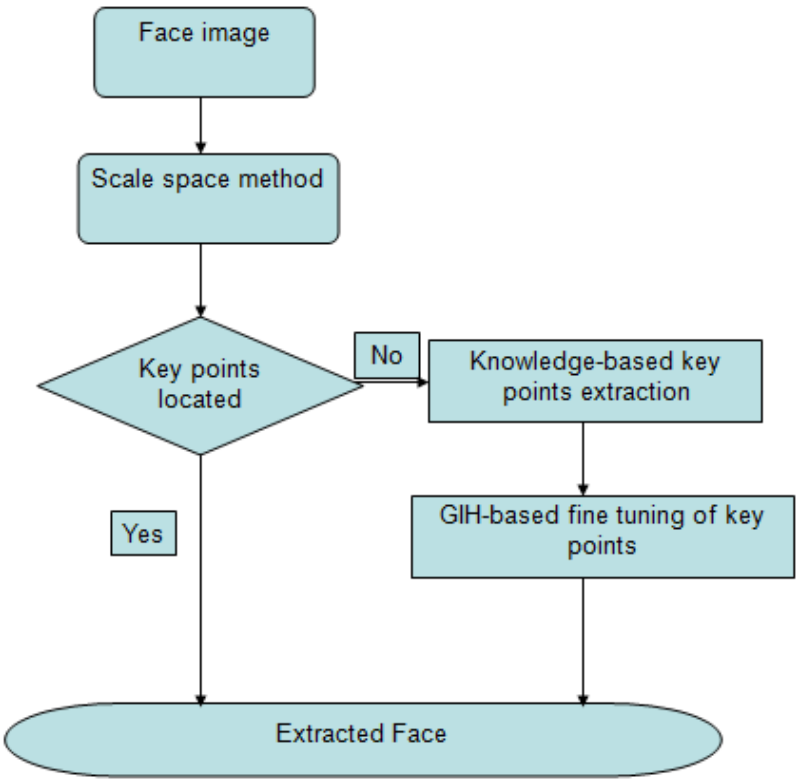

Figure 14, Flow diagram of face extraction [15]

3.1.1 Scale-space Method

The goal of this method is to find a simple and fast approach for locating faces in profile images [12]. First, the image is converted to a binary black and white image. A pre-processing step then extracts the outline curve of the front of the silhouette. The profile line is converted from two-dimension to one-dimension line $f(x)$, where $f(x)$ presents the column index and x presents row index of a pixel. Profile line $f(x)$ is than smoothed and flattened with Gaussian convolution of some particular standard deviation parameter. The first derivative of smoothed $f(x)$ is calculated. Nose tip can be considered as an extreme of the profile line whose first derivative equals zero. After we select all the extrema in the profile line, we can locate the unique nose tip based on some criteria. Based on nose tip, one can extract the face from background.

The performance of this method for profile face extraction is robust. One example is shown in Figure 15. The blue point indicates the position of nose tip. Yong et al. [15] showed that with selecting appropriate σ parameter, this method can not only work with profile images but also face of different angles. We show the result with different poses in Figure 16, and the yellow cross indicates the eye edge.

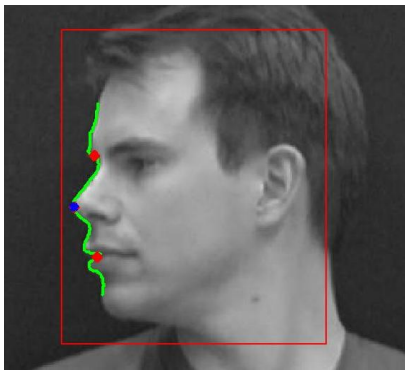

Figure 15, Nose tip location

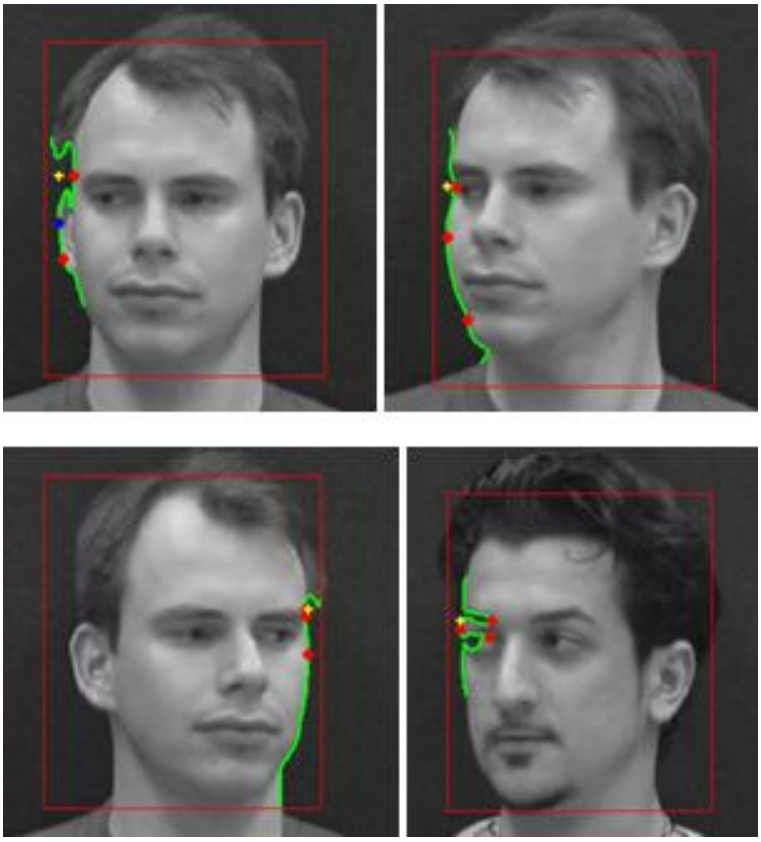

Figure 16, The key point location at eye edge.

Note that the eye edge point can be located with this method for different angles and different persons. Although the nose tip cannot be located in image examples shown in Figure 16, the eye edge points can be used to extract face instead of nose tip. Therefore, we improve this method to make it more useful for all pose angles. The main modifications to the algorithm from Figure 4 are as follows: After obtaining the set of key points, we check if there is a key point such as a nose tip or eye edge. If these key points exist, our algorithm then extracts faces using one of these key points such as nose tip or eye edge, respectively.

1. Extract the profile line f(x) from input image.

2. Smooth the profile line using Gaussian convolution with small parameter σ_s;

$$F(x, \sigma_s) = f(x) * g(x, \sigma_s)$$

3. Smooth the profile line using Gaussian convolution with large parameter σ_l;

$$F(x, \sigma_l) = f(x) * g(x, \sigma_l)$$

4. Flatten the profile line; $ff(x, \sigma_s, \sigma_l) = F(x, \sigma_s) - F(x, \sigma_l)$

5. Compute Gaussian convolution with σ_n;

$$FN(x, \sigma_n, \sigma_s, \sigma_l) = ff(x, \sigma_s, \sigma_l) * g(x, \sigma_n)$$

6. Locate extrema in $FN(x, \sigma_n, \sigma_s, \sigma_l)$.

7. Use criteria $((a_1 > 0) \,\&\, (a_2 < 0) \,\&\, (b_1 > b_2))$ to locate nose tip.

8. Classify the key points, find the nose tip or eye edge respectively.

Figure 17 shows the improved algorithm for key point location [15].

3.1.2 Facial Structure and GIH

The scale-space method achieves good face extraction for rotated faces. However, this result falls short on frontal images. As shown in figure 18, if the profile line is too smooth or jagged, we cannot extract extrema to locate the key feature points.

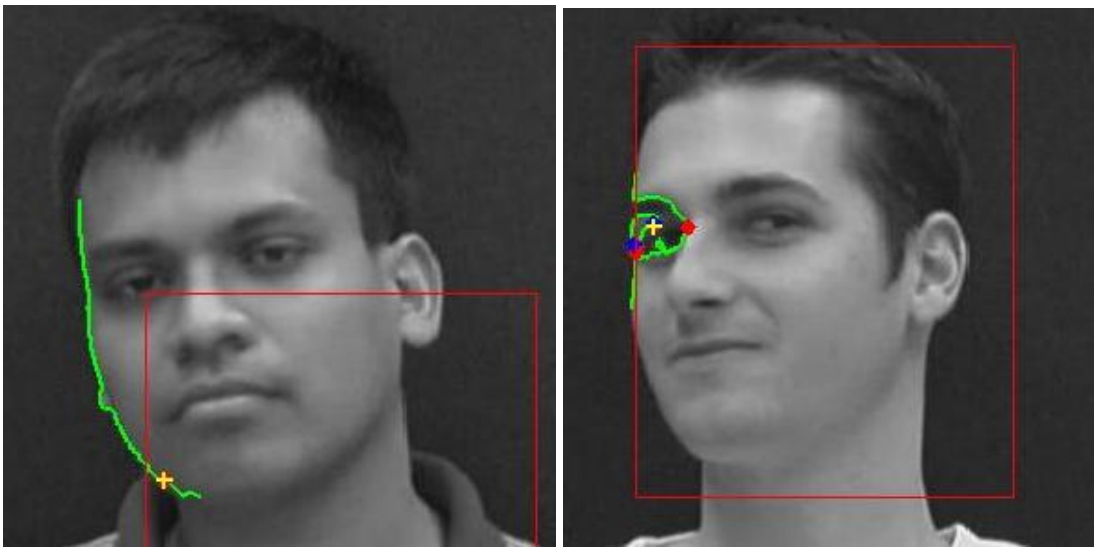

Figure 18, scale-space method fails for frontal and slightly rotated faces.

From our flow diagram in Figure 14, when this method fails we proceed by the facial structure knowledge method based on the location of the two irises. Firstly, the image is converted into binary image, and some pairs of connected areas are founded. By using the face structural knowledge, a particular pair of connected area L and R where the two irises locate can be identified. The criterions are listed below [18]:

A. The L and R should be in the middle part of the image;

B. There is no connected area under the L and R in a certain distance;

C. The distance between the vertical coordinate of center points of L and R must be less than a certain value;

D. The distance between the horizontal coordinate of center points of L and R must be greater than a certain value.

Based on the location of the irises, we extract face from the background. Figure 19 shows one example of locating irises and face using facial structural knowledge method.

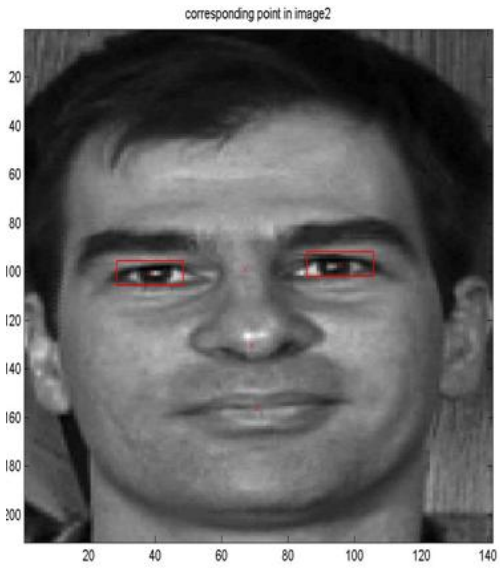

Figure 19, Key points locating based on face structural knowledge.

After locating the two key points of eye, Yong et al. [15] employ a GIH method to improve the accuracy of key point location. Geodesic-intensity histogram (GIH) discussed in Chapter 2, is a deformation invariant descriptor extracted from geodesic sampling. It captures the joint distribution of the geodesic distance and the intensity of the sample points. We can use the deformation invariant property of GIH to identify corresponding points. After the key points of eye are located by face structural knowledge method, we set a 20 by 20 mask around each key point. A template, which contains the correct position of eyes, is used as a marker. The GIH of each point in the mask are calculated, and they are compared with the GIH of the marker. χ^2 , defined in equation 2-7, is used to describe the similarity of two GIHs. The smaller χ^2 the more similar they are. Therefore, the point within the mask with the smallest χ^2 is selected as the new matching key point. Figure 20 shows the overall algorithm.

1. Convert image to binary image.
2. Locate position of two irises based on the structural knowledge.
3. Set a 20 by 20 mask around each key point.
4. Compute similarity χ^2 between each point in the mask and the marker in template.
5. Select the point within the mask with the smallest χ^2 as the new matching key point.
6. Extract face based on the new key point

Figure 20, The algorithm for facial structural knowledge and GIH [15].

In addition to the face extraction, these methods are used to obtain the location of feature points in the images. Since we deal with both frontal and rotated views, some points are invisible for different poses. The only dependable features that are invariant to rotation are the mid-eye (point between the eyes), the nose tip, and the mouth. The coordinates of these points are extracted with the face.

3.2 Feature Extraction

Several emotion recognition systems rely on detecting small changes at key locations in a face [29]. Such systems track several feature point on a face frame in a video sequence, and compute the changes from frame to frame. These changes can be represented as vectors known as Action Units (AU). In [30], Cohen et al. also compute Motion Unit (MU) vectors as shown in Figure 21. They are similar, however not equivalent to AU's and are numeric in nature. They represent not only the activation of a facial region, but also the direction and intensity of the motion.

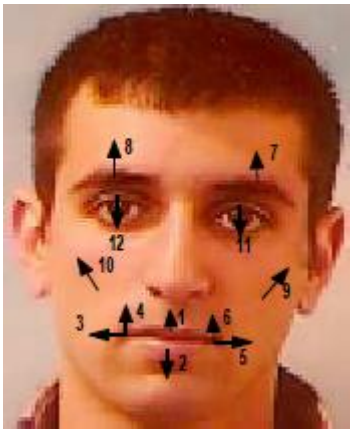

Figure 21, Example of Motion Units on a face [29]

MU's can be calculated from the optical flow between successive frames of a sequence. Optical flow is the distribution of apparent velocities of movement of brightness patterns in an image [30]. As detailed in chapter 2, the HS algorithm calculates the optical flow for each pixel in an image, as the change in brightness from that picture to the next. We implement this algorithm to compute the flow for each pair of successive frames. Figure 22 represents the flow patterns correctly portraying a left to right (right to left in picture) movement.

To obtain MU's the flow must be determined at the locations of feature points (mid-eye, nose tip, mouth). To make room for some error with the exact location of the points, we determine the flow to a small neighborhood using an averaging mask.

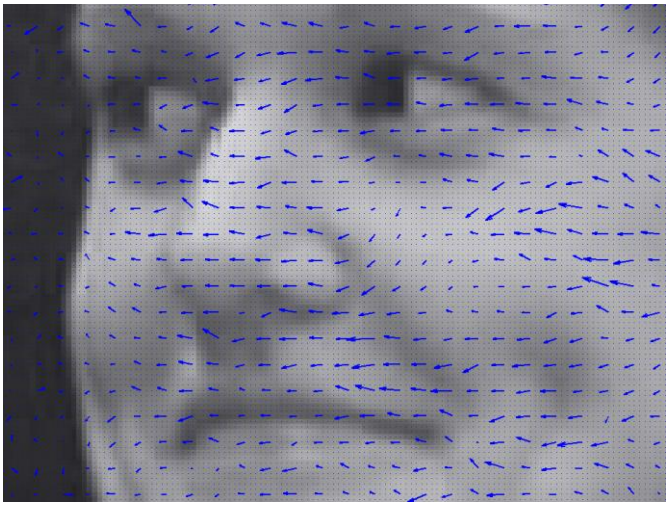

Figure 22, Optical flow movement patterns correctly show the face rotating right (right to left in picture)

3.3 Identification

As shown in [15], Yong et al. was the first to employ the CSRN in face image processing. The authors proved that the CSRN is a powerful tool to learn and predict temporal signature in image sequences, better than an RNN such as the Elman network. They trained the network with multiple but identical image sequences that follow the same motion. In this thesis our goal is to go beyond that and employ the CSRN for large-scale face identification problem with varying pose sequences. The sequences we use change in pattern (example: left to right, then right to left) and are also taken at different times. Motion units that capture the movement of the key points are needed to learn the sequence patterns.

Several steps are involved in the identification process. We start with a PCA step based on Turk and Pentland [5], to obtain eigenfaces. For each person, we create an eigenface from the training images. Then we project the eigenvector of each face to the eigenface subspace. The result is a vector representation of the image or face called

pattern vectors. However, using them as such is very computationally expensive. We can reduce the dimensionality by using PCA, thus retaining the first 9 principal components. These 9 dimensional reduced pattern vectors are subsequently combined with the MU's to form 10 dimensional pattern vectors.

As in [15] we formulate our pose variant recognition as a temporal prediction problem in image sequences. "Temporal Signature" is the difference between successive frames. If we use the pattern vector Ω to represent an image frame, the Euclidian distance between two successive frames is given as, $\varepsilon = \|\Omega_t - \Omega_{t+1}\|$; where t is the frame index. We consider this distance ε as the temporal signature in a face class. The CSRN takes the pattern vectors as input to learn and predict the temporal information. One CSRN represents one person and is trained by the corresponding sequences of this person.

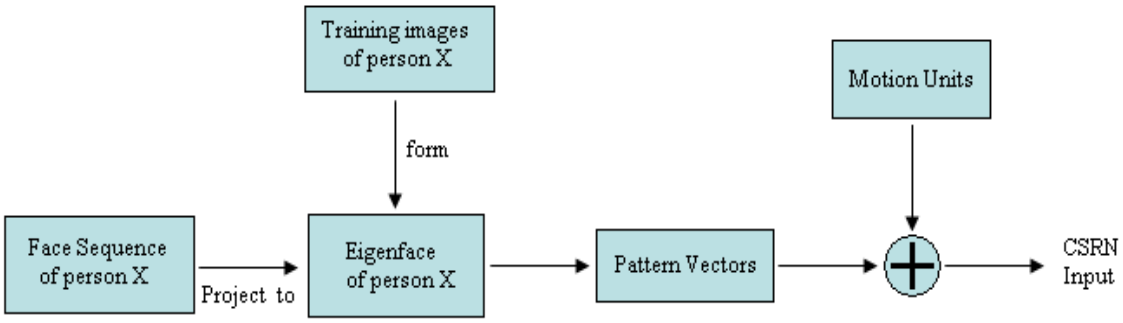

Figure 23, Data input for training.

Because one CSRN only has one input and one output and each pattern vector has 10 elements, in order to deal with one pattern vector at one time, we obtain 10 CSRNs as a group for one pattern vector. Therefore, each group of CSRNs represents one person. Each CSRN group is trained by corresponding pattern vectors, and each CSRN group

learns how to associate one face class/person in the training phase. For ease of reference describe, we mention one CSRN to mean one group of CSRNs in the following contents. After training, one CSRN is associated with one person. When a new testing sequence is encountered, this sequence is projected to all eigenface subspaces of different person to obtain pattern vector sequences. Then, each test pattern vector sequence is applied to the specific CSRN group associated with the corresponding face class. This process is shown in Figure 24.

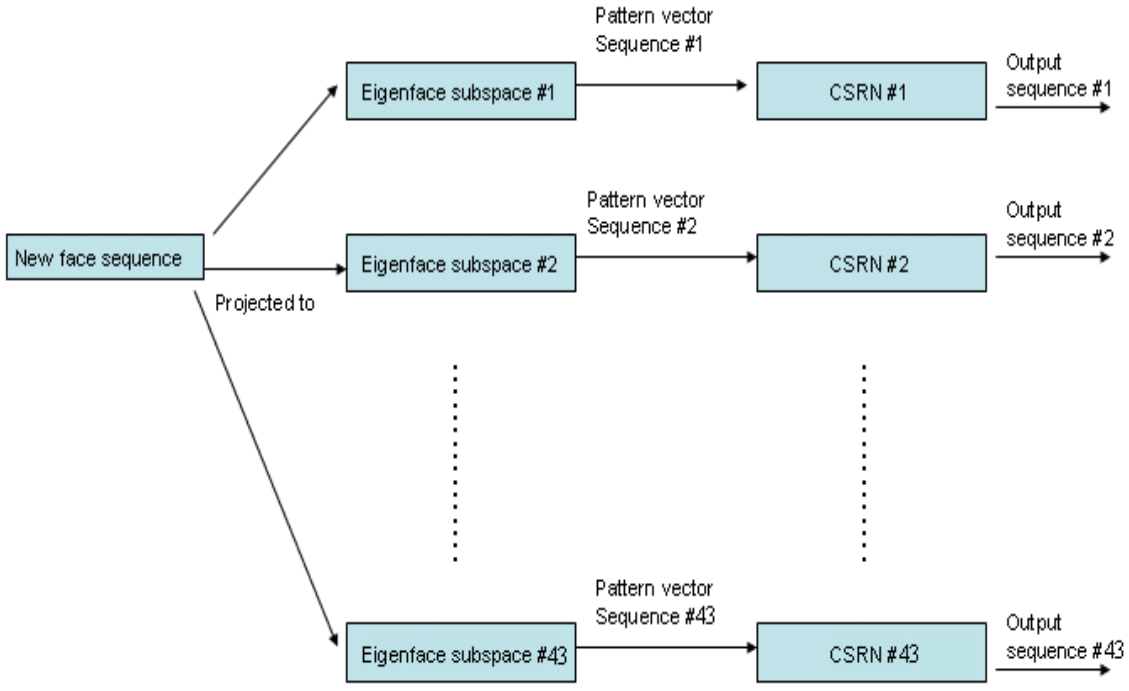

Figure 24, Testing steps

Now, we have one output sequence and one input sequences for each CSRN. Two classes of temporal Euclidian distance are computed as shown in Figure 25; one is the distance between successive frames of input sequence (solid blue lines), and the other is

the distance between successive frames of output sequence (dotted red lines). If the test sequence is from the corresponding person, then the input distance and the output distance may be very similar. Based on the similarity of lines, we can identify corresponding person.

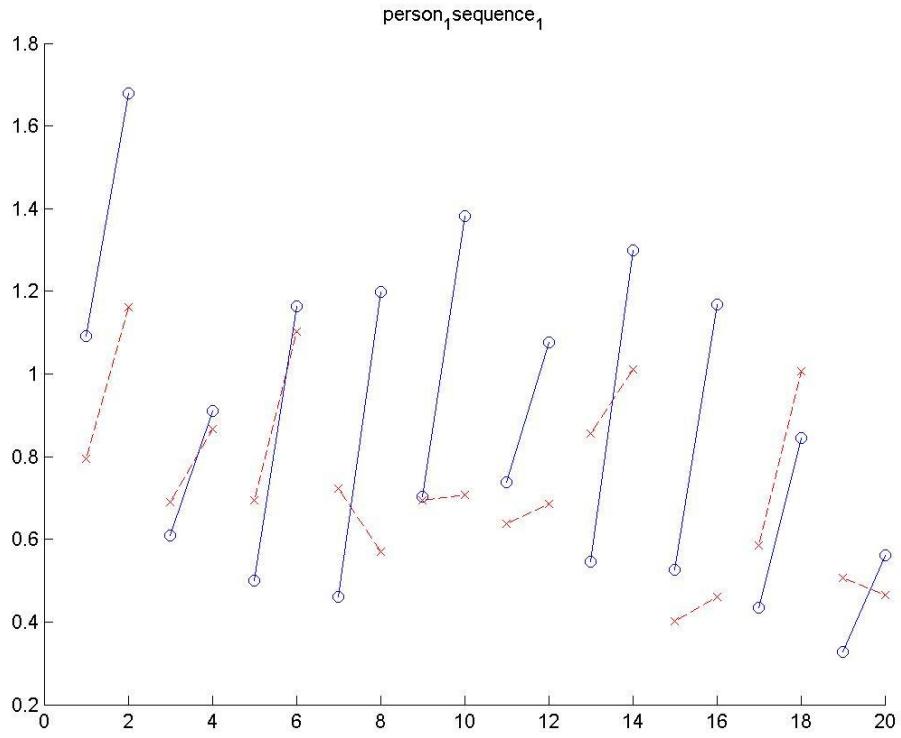

Figure 25, The Euclidian distance of person 1 sequence 1

In order to evaluate the similarity of two lines in a pair, we introduce an automated method based on the slope, distance, and convolution of these two lines [15]. Here we define ξ as the similarity of the two lines,

$$\xi = 0.6 * dist1 + 0.5 * dist2 + 0.5 slope1 + 0.2 * slope1 - 0.2 * conv \quad (3.1)$$

In this equation, dist1 is the vertical distance between the first point in a line of our output and the corresponding first point in our goal. Dist2 works similarly for the third point. Slope1 is the difference between the slope of our output and the slope of the goal for the first half of a line. Slope2 works likewise. We subtracted the convolution of the two lines since it is a measurement of similarity as opposed to our other measurements that are measurements of differences. With this automated method, the set of lines with the lowest ξ are our most similar lines.

3.4 Evaluation

In this thesis, we propose a new evaluation method for the CSRN based on the FERET evaluation detailed in Chapter 2. The results of our simulation are compared with standard algorithms from the CSU toolbox [28]. The algorithms implemented are the four main method used in the FERET and FRVT test: PCA from [5], BIC (Bayesian intrapersonal/Extrapersonel Classifier) based on [6], LDA from [7], and EBGM from [8]. The CSU toolbox has been developed as a reference and a comparison tool for different researchers to use. More specifically the four algorithms deal with identification only, they are semi-automatic, the user must provide the coordinates of key point in the face, and they are aimed for static recognition. Therefore the CSU toolbox needs to be modified to deal with the feature coordinates, and with making the system dynamic. We introduce a few additional steps to the provided algorithms and employ the pre-processing method described in section 3.1 to extract the coordinates of key points.

As explained by the FRVT performance metrics [27] and the CSU manual [28], a face recognition algorithm must produce a $N \times N$ similarity matrix, which contains the distance between all N images considered. This matrix is used to find the distance

between images in the gallery G and the probe list P_g, then the identification rate is computed using Equations (2.30) and (2.31). A dynamic system must produce a similarity matrix that contains the distance between sequences not images. To achieve a dynamic identification in video, we implement a few steps to convert the similarity matrix from images to sequences. Similarly the CSRN also produces a sequence similarity matrix. Figure 26 shows the execution flow of the four algorithms with the additional steps.

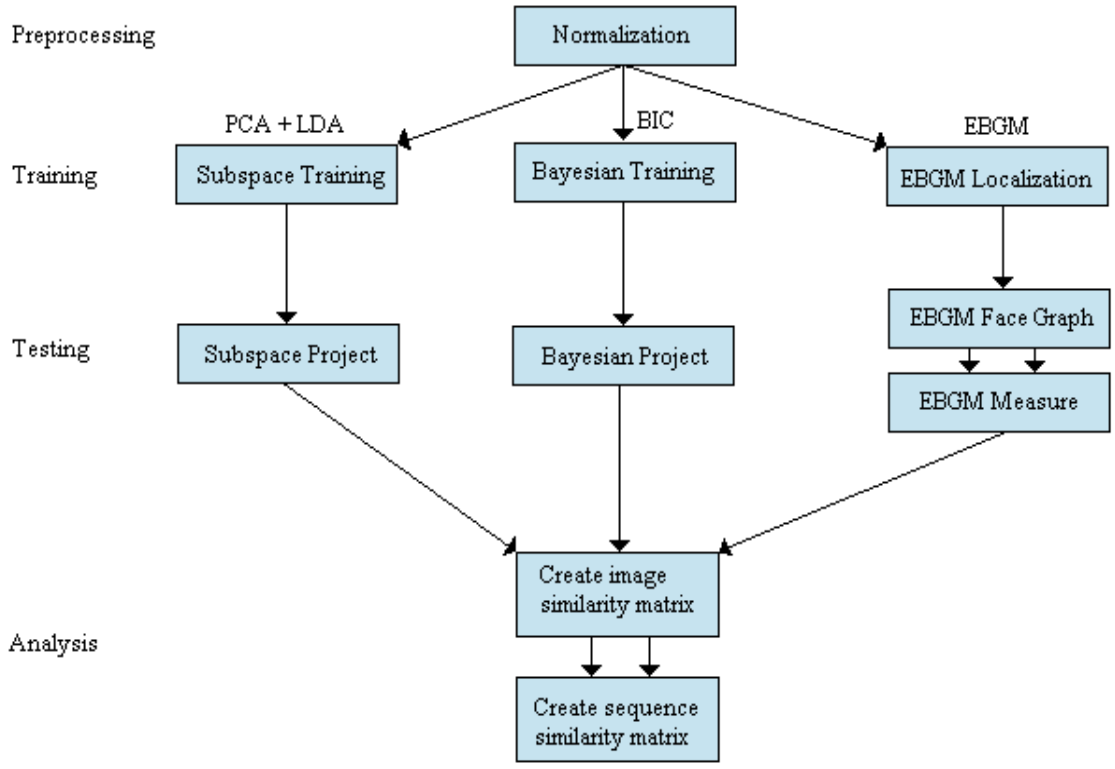

Figure 26, Flow of the modified CSU algorithms

We discuss the detailed steps in the following. Consider a simple example of N total images, divided among p persons. For simplicity, all persons have an equal number of 6 images although this is not required. In a dynamic case the images or frames are organized by sequences and each person has s sequences of f frames for a total of n

images. Then $s \times f = n$ and $p \times s \times f = N$. Figure 27 illustrates this with $n = 6$ total images and $s = 2$ sequences per person, and then $f = 3$ images per sequence. Each cell represents the distance Xi-j between image I and j. The $N \times N$ matrix is organized in such a way that each row and column sequentially represents the ordered images of person 1, followed by the ordered images of person 2, and so on. Therefore, the first f rows or columns represent the first sequence of person 1. The following f rows or columns represent the second sequence of person 1. Hence the similarity matrix is a block matrix of $p \times s$ blocks where each block represents a sequence, and the distance between the blocks is the distance between sequences as shown in Figure 27.

			person 1						person 2						.	.	.	.
			sequence 1			sequence 2			sequence 1			sequence 2			.	.	.	.
			IM 1	IM 2	IM 3	IM 4	IM 5	IM 6	IM 7	IM 8	IM 9	IM 10	IM 11	IM 12	.	.	.	IM n
person 1	sequence 1	IM 1	X1-1	X1-2	X1-3	X1-4	X1-5	X1-6	X1-7	X1-8	X1-9	X1-10	X1-11	X1-12				.
		IM 2	X2-1	X2-2	X2-3	X2-4	X2-5	X2-6	X2-7	X2-8	X2-9	X2-10	X2-11	X2-12				.
		IM 3	X3-1	X3-2	X3-3	X3-4	X3-5	X3-6	X3-7	X3-8	X3-9	X3-10	X3-11	X3-12				.
	sequence 2	IM 4	X4-1	X4-2	X4-3	X4-4	X4-5	X4-6	X4-7	X4-8	X4-9	X4-10	X4-11	X4-112				.
		IM 5	X5-1	X5-2	X5-3	X5-4	X5-5	X5-6	X5-7	X5-8	X5-9	X5-10	X5-11	X5-12				.
		IM 6	X6-1	X6-2	X6-3	X6-4	X6-5	X6-6	X6-7	X6-8	X6-9	X6-10	X6-11	X6-12				.
person 2	sequence 1	IM 7	X7-1	X7-2	X7-3	X7-4	X7-5	X7-6	X7-7	X7-8	X7-9	X7-10	X7-11	X7-12				.
		IM 8	X8-1	X8-2	X8-3	X8-4	X8-5	X8-6	X8-7	X8-8	X8-9	X8-10	X8-11	X8-12				.
		IM 9	X9-1	X9-2	X9-3	X9-4	X9-5	X9-6	X9-7	X9-8	X9-9	X9-10	X9-11	X9-12				.
	sequence 2	IM 10	X10-1	X10-2	X10-3	X10-4	X10-5	X10-6	X10-7	X10-8	X10-9	X10-10	X10-113	X10-12				.
		IM 11	X11-1	X11-2	X11-3	X11-4	X11-5	X11-6	X11-7	X11-8	X11-9	X11-10	X11-11	X11-12				.
		IM 12	X12-1	X12-2	X12-3	X12-4	X12-5	X12-6	X12-7	X12-8	X12-9	X12-10	X12-11	X12-12				.
.	.	.													.			.
.	.	.														.		.
.	.	.															.	.
.	.	IM n	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	Xn-n

Figure 27, $N \times N$ Similarity matrix where each $f \times f$ (in this case $f = 3$) block represents a sequence.

We can convert the $N \times N$ image-to-image to a $N/f \times N/f$ sequence-to-sequence similarity matrix by applying a simple weighted $f \times f$ averaging mask that reduces each $f \times f$ block to a single distance value. Then instead of considering gallery and probe

images, we look into gallery and probe sequences and form the similarity measure around them. The CSRN will also produce an identical matrix to be used for determining similarity between sequences.

4 Experimental Results

We conduct all our experiments with the VidTimit dataset [26]. It contains video sequences of 43 persons. Each person performs a head rotation starting from frontal view. We selected samples from each face rotation to form sequences of 4 images or frames. Each sequence represents one movement (e.g. 0 to -90, -90 to 0, 0 to +90). Some sequences are neutral and contain no rotation; these will be used as gallery sequences. An example is shown in Figure 28.

Figure 28, Example of 5 face sequences.

For each person we select 9 sequences of 4 frames for a total of 36 frames per subject and 1548 images total. In out experiment we use only 4 selected frames to represent the movement. The original dataset contains more frame samples showing the movement continuously.

4.1 Face and Feature Extraction

In [15] Yong et al. reported an extraction rate of 89.2% using the scale-space method. The images that are not extracted properly are then pre-processed with the facial knowledge with GIH method with an extraction rate of 83%. This brings to a total of 98% correct extraction. Our pre-processing procedure is identical with the addition of feature selection for landmark points (mid-eye, nose tip, mouth), and Motion units computation at these landmark points.

Following the steps of scale-space method in Figure 15, all the images are converted to binary images. The profile line is extracted through edge detection and transformed from 2D to 1D. The extracted profile line is then smoothed and flattened, using Gaussian convolution of a function with specific standard deviation parameter. The first derivative of smoothed profile line is calculated. Subsequently, we select all the extrema whose first derivative equals zero, and test these extrema with the criteria explained in Chapter 3 to identify the unique nose tip, mid-eye edge point or mouth. Based on the key points, we can extract the face from background.

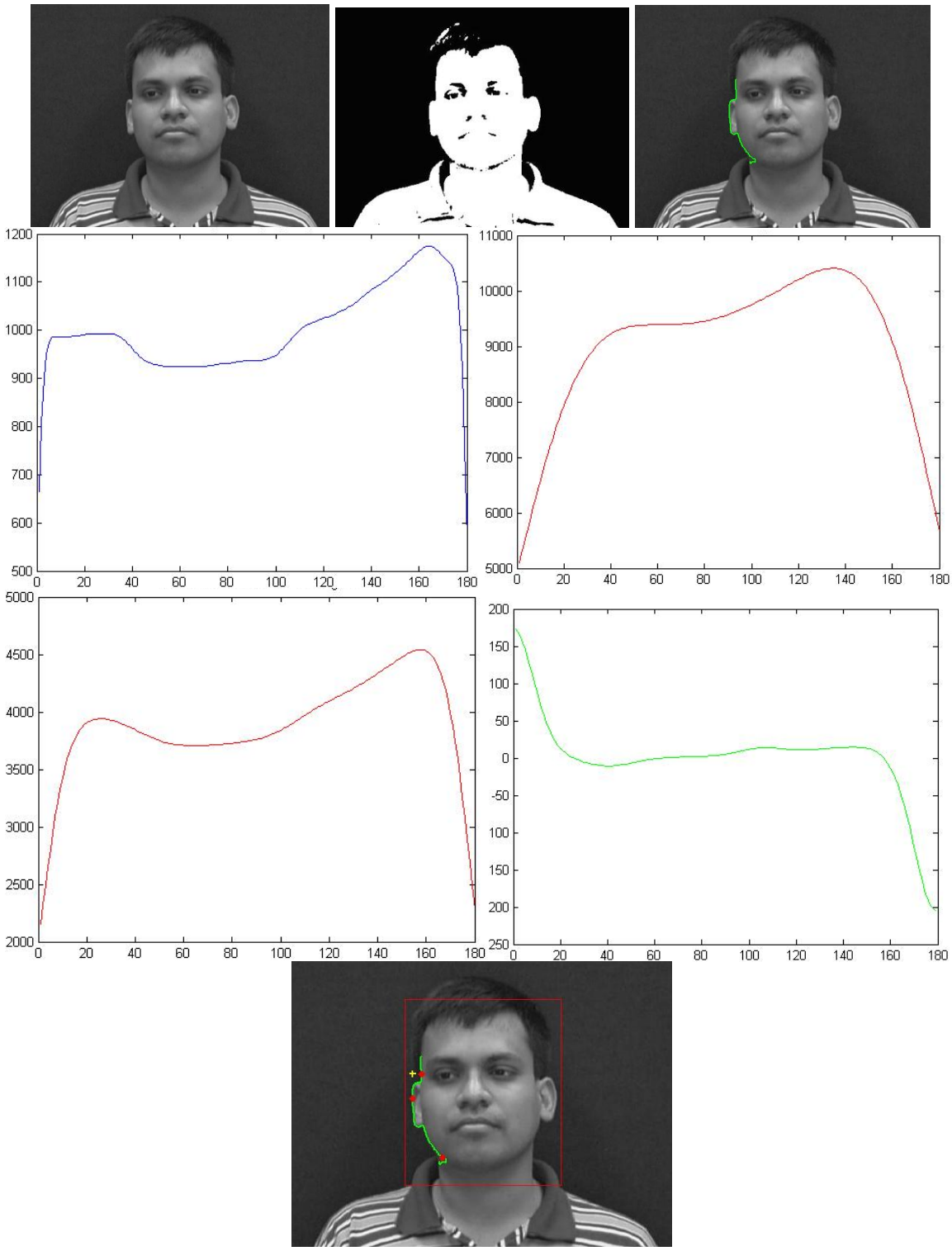

Figure 29, Processing and face extraction steps with scale-space method (a) Original image (b) Binary image (c) Extracted profile line (d) 1D line after 1st convolution (e) 1D line after 2nd convolution (f) Flattened line (g) First order derivative (h) face and feature extraction

Using the VidTIMIT dataset, the extraction rate for the scale space method is 86.3% all angles included. Figure 29 shows the step by step result for successful face extraction. However, we observe that even though the extraction worked, in some cases the feature points selected are inaccurate as shown in Figure 30 (b) and (c). The feature selection has a lower success rate of 59.5%. The images that could not be extracted successfully are mainly from frontal or near frontal view. This failure is due to two reasons; either the profile line is too smooth for locating extrema and key points or the profile line is too twisted.

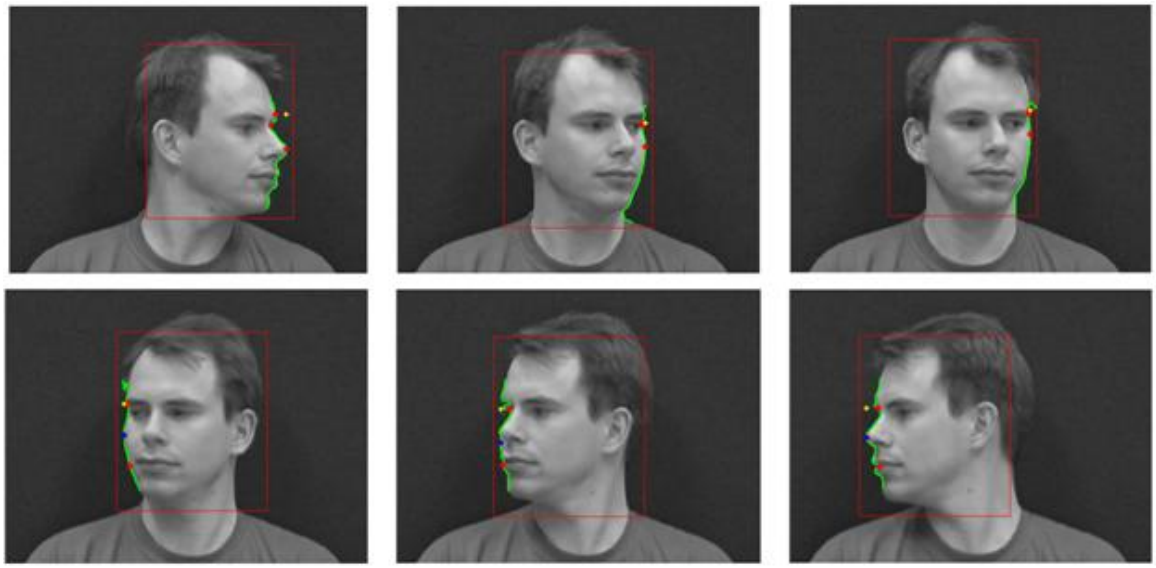

Figure 30, Face extraction with scale space method.

The images where the extraction and feature selection failed are pre-processed with the facial knowledge method. Again images are converted to binary forming some pairs of connected areas. We examine these connected areas to see if there is a particular pair of connected area L and R following the criteria detailed in Chapter 3. If we can

locate just one such pair, this is considered as the location of the irises. Then, we extract face from the background based on two irises. Figure 31 shows the example steps of locating irises and face.

Figure 31, (a) Original image, (b) Binary image, (c) Labeled binary image, (d) Eye position in the image, (e) Face extraction.

Note in Figure 31 (e) the point locations are not very accurate. This can be improved by coupling with the GIH method. We set a 20 by 20 mask around the key points and compute the similarity χ^2, defined in Equation (2.7), of each point in the mask. Therefore, the point within the mask with the smallest χ^2 is selected as the new matching key point. In Figure 32 (a) we show an extracted image, and the relocation of key point in 32 (b). The complete results for face and feature extraction using both methods is shown in table 2.

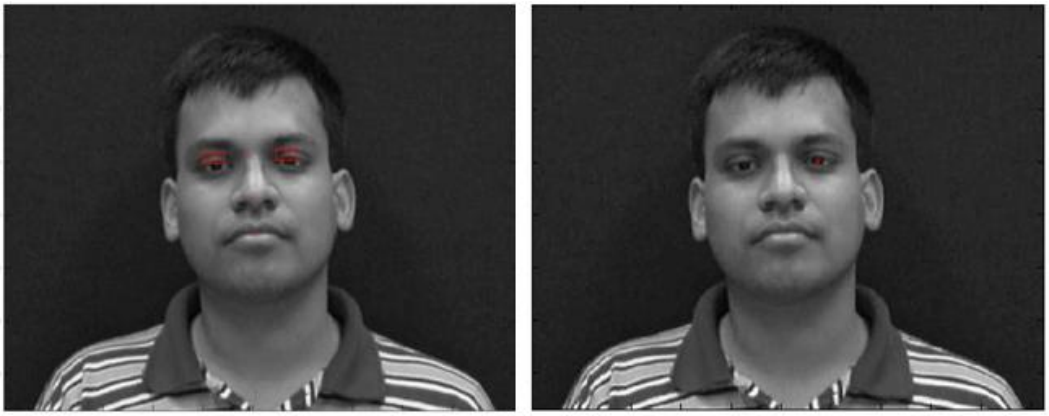

Figure 32, (A) original image, (B) the point moving to the correct position after GIH

Table 2: Face and Feature Extraction Results

Method	Correct Face Extractions	Face Extraction Rate	Correct Feature Extraction	Feature Extraction Rate
Scale-space	1336/1548	86.3%	921/1336	59.5%
Structural Knowledge	121/212	57%	171/627	41.6%
Total	1457/1548	93.5%	1182/1548	76.6%

4.2 Identification

In our face recognition part, we formulate our face recognition as a temporal prediction problem in image sequences, and use the CSRN to learn and predict temporal signature. After prediction we find the similarity between sequences. Each person has 9 sequences of 4 frames as shown in Figure 28. The first 6 are used for training and last 3 for testing. Our identification is based on how close the predicted testing (or probe)

sequences are to the gallery sequences. We perform 2 experiments. The first one is similar to the experiments done by Yong et al. [15], where all the sequences are similar but with a much larger dataset and we evaluate the results using the FERET evaluation system. For the second we introduce variations in the sequences and examine how motion units combined with our pattern vectors help improve results by detecting the direction of movement.

Figure 33, Input sequence

4.2.1 Experimental Procedure

Forty three subjects are considered with 9 sequences each, 6 sequences are for training and 3 for testing. The 24 training images (4 frames for each sequence) are used to create eigenface sub-spaces of each person. As discussed in Chapter 2, each image can be represented by a pattern vector; therefore 24 pattern vectors are computed by projecting the 24 images to the eigenface subspace. We can use a part of principal components elements in one pattern vector to approximate the representation of one image. So, in our experiments we choose the first 9 principal components of one pattern vector to represent one image. The motion unit is added as the 10^{th} dimension forming 10-dimensional pattern vectors as input to the CSRN. As described in Chapter 3, we compute the optical flow around the 3 feature points that are rotation invariant (e.g. mid-eye, tip of nose, and

mouth center). The coordinates for these points are provided from our pre-processing steps. A mask averages the motion around these points and obtains a motion unit vectors. By adding motion units, the pattern vectors will not only contain global information about the image but also information about local movement.

We construct 43 CSRNs. One CSRN is trained by corresponding pattern vectors from one particular person, and each CSRN is associated with one face class/person in the training phase. Now we have 43 trained CSRN groups, we use NN1, NN2 …NN43 to refer to the CSRN of the face class.

When a new testing sequence is encountered (e.g. a new testing sequence from person 1) this sequence is projected to all the 43 eigenface subspaces to obtain 43 test pattern vector sequences. Then, each test pattern vector sequence is applied to the specific CSRN group associated with the corresponding face class. For example, the pattern vector sequence obtained from person 1's eigenface subspace will be applied to NN1 and so on. This process is shown in Figure 34.

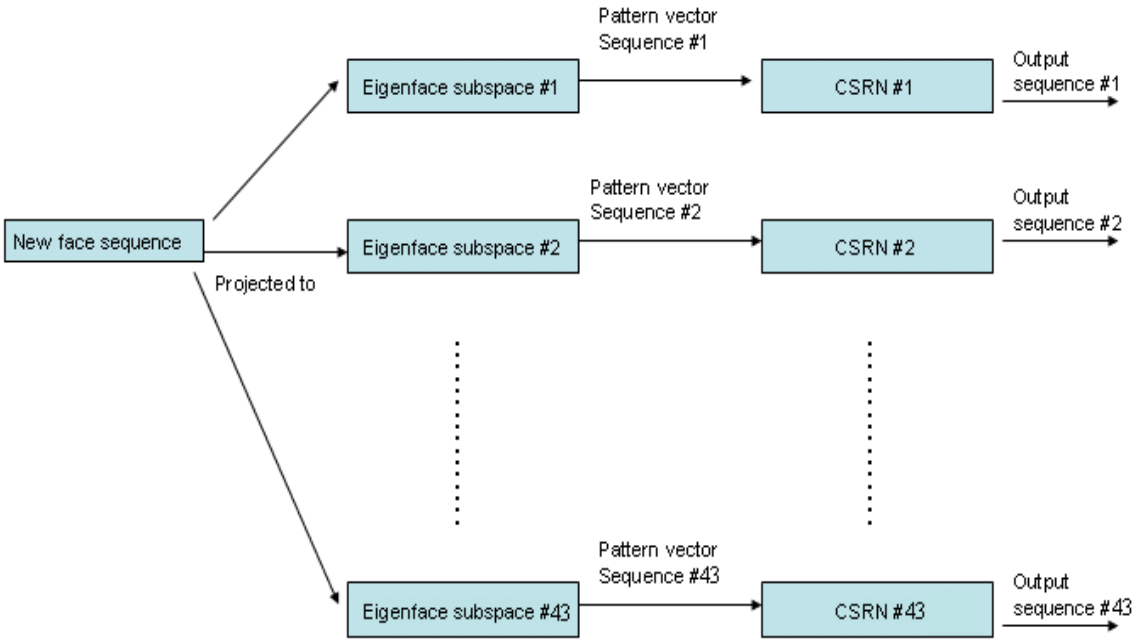

Figure 34, Testing process

Now, we have 43 output sequences corresponding to 43 input sequences. Two classes of temporal Euclidian distance are computed as shown in Figure 34; one is the distance between successive frames of input sequence (blue lines), and the other is the distance between successive frames of output sequence (red lines). If the test sequence is from the corresponding person, then the input distance and the output distance may be very similar. In Figure 35 we observe that the 6^{th} pair of lines appears very similar suggesting a match. This plot shows that the temporal signature can be used as a measure for recognition of large scale pose variant face images. For completeness, we show another example in Figure 36 where the 7^{th} pair of lines appears very similar suggesting a match with person 7, however the correct match should be person 4, therefore this is a mismatch.

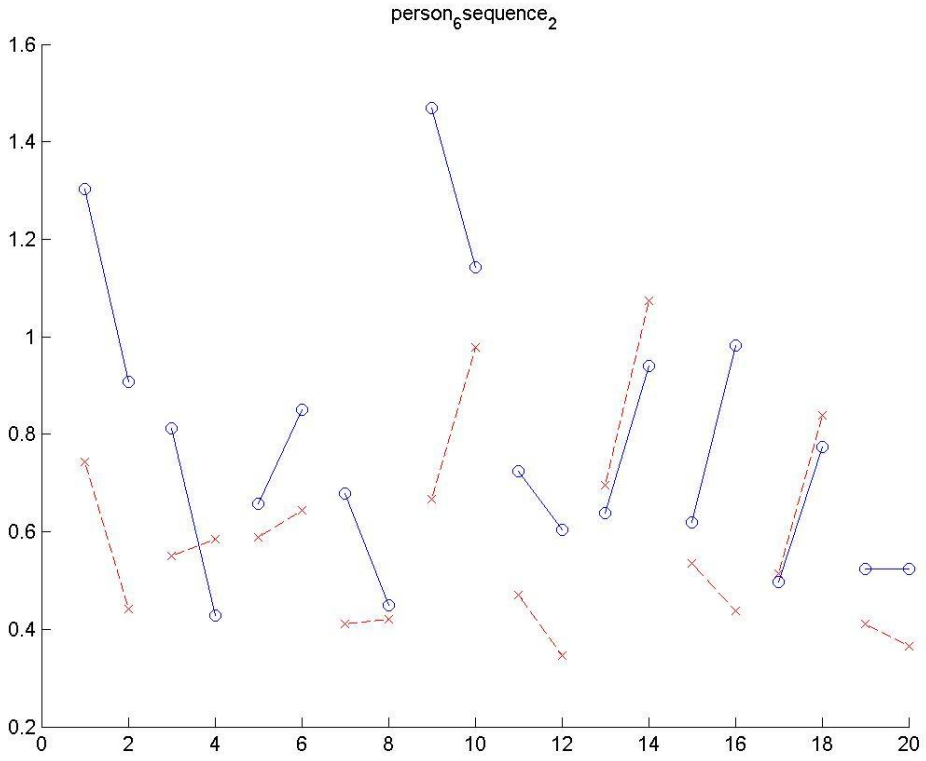

Figure 35, The Euclidian distance of person 6 sequence 2

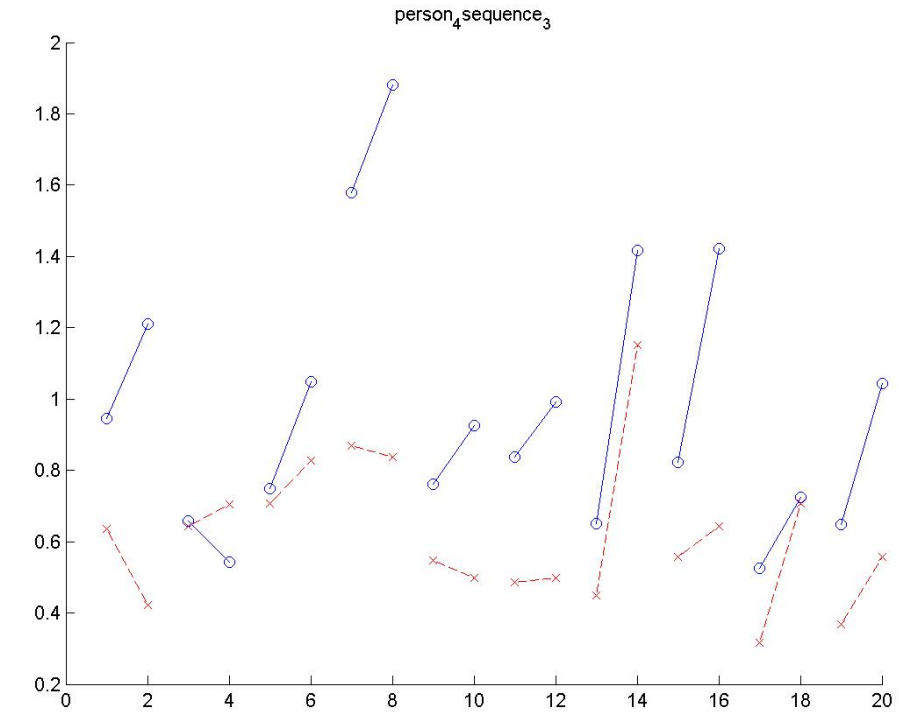

Figure 36, The Euclidian distance of person 4 sequence 3

In order to evaluate the similarity of two lines in a pair, we use the automated discriminating method proposed in Chapter 3. This method is based on the slope, distance, and convolution of these two lines. The similarity between lines will represent the distance between the sequence considered and every other person. So for every person we get the similarity between each of 3 testing or probe sequences considered and every other person. From here we build a sequence similarity matrix to be used for identification. In our thesis, the identification rate introduced in FRVT 2002 [2] is used as a universal metric to evaluate the performance of a face recognition system.

Experiment 1

The first experiment aims at demonstrating not only that the CSRN can learn and predict the temporal information in a sequence as was shown in [15], but applying this system for large-scale face identification. Each person has 9 sequences that follow the same gradual rotation pattern as shown in Figure 33. The face rotates from frontal 0^0 to -90^0, covering the entire right hand orientation. With 43 persons and 3 test sequences per person we obtain a total of 129 probe or test sequences. For this test set with CSRN, the overall identification rate is 46 correctly recognized sequences at rank 1 (46/129 = 35.66%) and 98 at rank 2 (98/129 = 76%). For large scale comparison and evaluation, we obtain the same face recognition performance using a few well known methods. The result at rank 1 is not the best. However, the CSRN identification shows increased rates at higher ranks. Table 3 shows the complete results up to rank 5, and Figure 37 show the corresponding rank curve for all these techniques. From these results we can conclude that the CSRN is more robust for one-sided (0^0 to -90^0) pose invariant face recognition using a large database when compared to standard techniques.

Table 3: Experiment 1 Identification results

Method	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5
CSRN	35.66%	47.29%	60.47%	67.44%	76%
PCA	31.4%	41.87%	55.81%	59.30%	70.1%
LDA	34.88%	52.32%	59.30%	63.96%	68.60%
Bayesian ML	34.88%	40.7%	43.02%	46.51%	53.49%
Bayesian MAP	30.23%	36.05%	38.37%	41.86%	46.51%
EBGM	38.76%	46.51%	48.83%	48.83%	49.61%

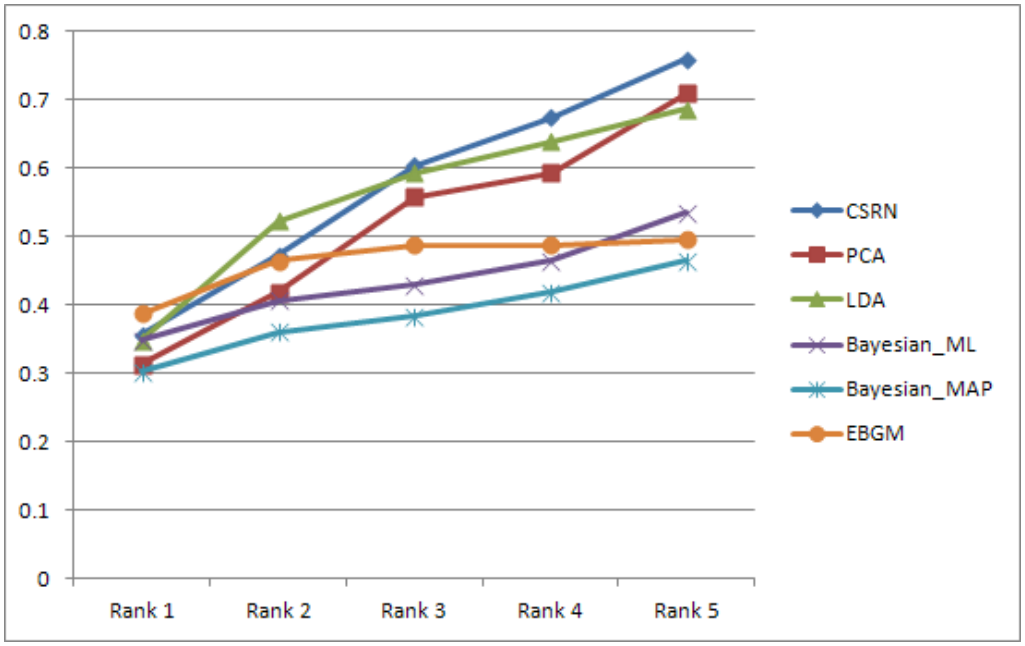

Figure 37, Experiment 1 rank curve

4.2.2 Experiment 2

In this second experiment, we go further in our study and evaluate the performance of our system with more complex pose variations. As shown in Figure 28, the sequences vary randomly in their direction and in the time the images were captured. The first attempt was to redo the same experiment as done by Yong et al. [15] with pose sequences similar to Figure 28, but without considering motion unit information in the feature vectors. The preliminary results obtained showed that the CSRN performed poorer with an identification rate of 22.6%, when compared to a standard algorithm such as EBGM with 34.88% at rank 1. This was due to the fact that the CSRN was unable to learn pose variations with multiple directions of movement. Therefore we introduced motion unit vectors that describe the movement and local variation of facial features, and incorporated them with our input pattern vectors.

As in experiment 1, 24 pattern vectors are computed by projecting the 24 training images to the eigenface subspace for each person. We use the same set of 43 persons with 9 sequences per person, 6 training sequences and 3 for testing. Among all the 129 probe or test sequences for 43 persons, the overall identification rate is 39 sequences recognized correctly at rank 1 (39/129 = 30.23%) and 100 at rank 5 (100/129 = 77.51%). Similar to experiment 1, the CSRN identification has an average performance at rank 1 but the rate increases rapidly at higher ranks. Using the same dataset we evaluate the other face recognition techniques with our modified toolbox. The overall evaluation results shown are in Table 4 and Figure 38.

Table 4: Experiment 2 Identification results

Method	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5
CSRN	30.23%	52.78%	66.67%	73.64	77.51%
PCA	31.4%	38.37%	50%	55.81%	61.63%
LDA	32.56%	52.33%	59.30%	63.95%	67.44%
Bayesian ML	25.58%	30.23%	36.05%	38.37%	40.70%
Bayesian MAP	30.23%	36.05%	38.37%	44.19%	51.16%
EBGM	34.88%	41.86%	45.35%	50%	58.14%

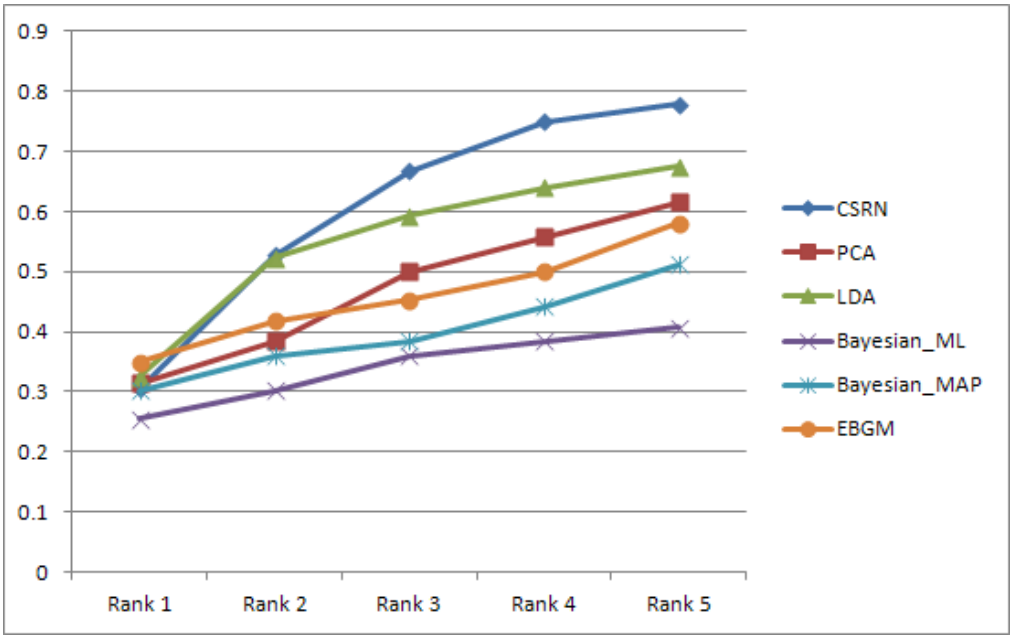

Figure 38, Experiment 2 rank curve

5 Conclusions

5.1 Summary

Several face recognition techniques have been proposed to deal with large rotation. However, as discussed in FRVT 2002 [3], the recognition rate for non-frontal images is low. Yong et al. [15] proposed a novel technique based on temporal information in facial image sequences and used a CSRN to obtain large-scale pose invariant face recognition. In this work, our goals are to i) fully evaluate the large scale face rotation technique in [15] using large pose variation database; ii) compare and benchmark CSRN performance with well know comparable face recognition techniques; and iii) enhance the CSRN system to solve the dynamic video identification problem.

Our fully automatic system consists of two steps, pre-processing (segmentation/feature selection) and identification. A scale-space segmentation [18] designed to work with profile images is applied. Yong et al. modified and improved this method to work with frontal or near frontal view. The scale-space method has an excellent performance for rotated faces and semi rotated faces with an extraction rate of 86.3% for all face images in the dataset. However, few frontal faces cannot be processed successfully due to the absence of key points. This method is combined with face structural knowledge which works well for frontal face. The GIH is also used to improve the accuracy of locating key points. The facial structural knowledge method is employed to deal with the images that couldn't be processed by scale-space, and achieves 57% extraction rate. Combining these two approaches, our system obtain an overall face extraction rate of 93.5%. In addition, we apply these same techniques to extract invariant point in faces. This feature extraction has a much lower result, and we obtain 59.5% correctly extracted features with the scale-space method. Again we apply the facial

structural knowledge method for those images with incorrectly extracted features, and perform extraction with 41.6% accuracy bringing it to a total of 76.6% correct feature extraction rate. At these feature points, we compute the optical flow based on successive sequence frames, and obtain motion unit vectors that describe the local movement in the sequence. In our identification phase, we formulate our face recognition as a temporal information prediction problem in image sequences. This formulation of face recognition allows us to use CSRN to learn and predict temporal signature. We exploit the same CSRN system used by Yong et al. in [15]. We use the eigenface technique to obtain pattern vectors that represent each frame and apply PCA to reduce their dimentionality. These pattern vectors combined with the motion units are our inputs for CSRN processing. The temporal distances of successive frames are calculated and plotted as the temporal information in a sequence.

The first goal for this research was to evaluate Yong et al.'s [15] original CSRN system by making it adaptable with large databases. With the original set up in [15], we could not automatically obtain the recognition performance using distance plots for a large number of sequences. With the modifications in this work, we can now obtain the distance plots between any number of test sequences based on the FERET standard similarity measure. We measure the similarity between input curves that represent gallery sequences and output curves that represent probe sequences and form a similarity matrix between sequences. The identification rate is calculated as defined in FRVT 2002 [3] by finding the number of matching subjects between a known gallery and a probe list based on a similarity matrix.

The second goal for this research was to be able to compare the CSRN system with well-know comparable algorithms for suitable benchmarking. The CSU face

identification toolbox offers implementations for PCA, Bayesian classifier, LDA and EBGM. This toolbox needed to be modified to perform exact comparison from image-to-image to sequence-to-sequence in a dynamic identification environment as disussed in this work. With the publicly available VidTIMIT Audio-Video face dataset, we measure the performance of our CSRN system with the standard algorithms for face image sequences with pose variations ranging from 90° to -90°.

The third goal for this research was to further enhance the CSRN system making it robust with variations in the database images and sequence patterns. The introduction of motion units to capture the local flow of movement in a sequence proved useful. In our first experiment, we demonstrate the ability of the CSRN with a large dataset and obtain an overall rate of 35.66% at rank 1. In the second experiment we test our system for handling modifications in the data sequences and obtain a 43.3% identification rate at rank 1 compared to a rate of 30.2% without motion units. Note our CSRN system shows increased recognition rate when compared to other well-known face recognition techniques at higher ranks. This improved performance suggests slow learning at rank 1 while retaining higher rates at higher ranks for CSRN compared to other methods including EBGM. We believe this retention in learning may be attributed to reinforcement technique for CSRN as discussed in [14] and [15].

5.2 Future Works

A few improvements can still be made for the current CSRN system. The similarity equation currently used to compute the distance between input and output sequences can be described as a weighted convolution formula. It works well in many cases when the number of images or samples in the sequence is small. It would be

preferable to find a more adaptable and mathematically eloquent equation for that purpose. We could also assume the sequence curves to be waveforms and base the similarity measure on cross-correlation which is similar in nature to convolution.

Another improvement can be made on the structure and the training process of the network itself. In the current cellular structure, the output passed from each cell or SRN to its neighbors is not the actual output of the cell. Reverting to the real output could improve convergence time. The training time can also be enhanced by using Unscented Kalman Filter instead (UKF) of the Extended Kalman filter (EKF).

The CSRN system can prove promising for several FR or other types of problems that require capturing and learning temporal information. One such an example is Emotion detection which is similar in essence to our work. Emotion recognition techniques are increasingly using video based sequences to detect the temporal pattern as well as the movement of Action Unit, similar to our motion units, across a sequence of frames.

REFERENCES

[1] W. Zhao, R. Chellappa, P. J. Phillips and A. Rosenfeld, "Face Recognition: A Literature Survey," *ACM Comput. Surv*. Vol. 35, No. 4, pp. 399-458, December 2003.

[2] P. J. Phillips, H. Moon, S. A. Rizvi and P. J. Rauss, "The FERET Evaluation Methodology for Face-Recognition Algorithms," *IEEE Trans. Pattern Anal. Mach. Intell*. Vol. 22, No. 10, pp. 1090-1104, 2000.

[3] P. J. Phillips and P. Grother, "Face Recognition Vendor Test 2002: Evaluation Report," www.frvt.org/DLs/FRVT_2002_Evaluation_Report.pdf.

[4] P. J. Phillips, P. Grother and R. Michaels, "Face Recognition Vendor Test 2002 performance metrics," 4th International Conference on Audio Visual Based Person Authentication, 2002.

[5] M. Turk and A. Pentland, "Eigenfaces for recognition". *Journal of Cognitive Neuroscience*. Vol.3, No.1, pp. 586-591, 1991.

[6] B. Moghaddam, C. Nastar and A. Pentland, "A Bayesian similarity measure for direct image matching," International Conference on Pattern Recognition. Vol. 2, 350, 1996.

[7] W. Zhao, R. Chellappa and A. Krishnaswamy, "Discriminant analysis of principal components for face recognition," International Conference on Automatic Face and Gesture Recognition. pp 336-341, 1998.

[8] L. Wiskott, J. M. Fellous and C. Von Der Malsburg, "Face recognition by elastic bunch graph matching," *IEEE Trans. Pattern Anal. Mach. Intell.* Vol. 19, No. 7, pp. 775–779, 1997.

[9] A. Lanitis, C. J. Taylor and T. F. Cootes, "Automatic face identification system using flexible appearance models," *Image Vis. Comput.* Vol. 13, No. 5, pp. 393–401, 1995.

[10] T. F. Cootes, G. J. Edwards and C. J. Taylor, "Active appearance models", *IEEE Trans. Pattern Anal. Mach. Intell*. Vol. 23, pp. 681–685, 2001.

[11] S. Zhou and R. Chellappa , "Image-based face recognition under illumination and pose variations", *Journal of the Optical Society of America A*. Vol. 22, No. 2, pp. 217-229, 2005.

[12] S. Gongy, A. Psarrouz, I. Katsoulisy and P. Palavouzisy, "Tracking and Recognition of Face Sequences," In European Workshop on Combined Real and Synthetic Image Processing for Broadcast and Video Production, Hamburg, Germany, 1994.

[13] J. Elman, “Finding structure in time”. *Cognitive Science*, Vol.14, No. 2, pp.179-211, 1990.

[14] R. Ilin, R. Kozma, and P. J. Werbos. “Beyond Backpropagation and Feedforward Models: a Practical Training Tool for a More Efficient Universal Approximator,” *IEEE Trans. Neural Networks*. Vol. 19, No. 3, 2008.

[15] Y. Ren, Khan M. Iftekharuddin, and W. E. White. “Large-scale Pose-invariant Face Recognition Using Cellular Simultaneous Recurrent Network,” *Applied Optics*. Vol. 49. No. 10, pp. 92-103, 2010.

[16] M. H. Yang, D. Kriegman and N. Ahuja, “Detecting faces in images: A survey,” *IEEE Trans. Patt. Anal. Mach. Intell*. Vol. 24, No. 1, pp. 34–58, 2002.

[17] H. A. Rowley, S. Baluja and T. Kanade, “Neural Network Based Face Detection,” *IEEE Trans. Patt. Anal. Mach. Intell.* Vol. 20, 1998.

[18] F. Jiao and W. Gao, “A Face Recognition Method Based on Local Feature Analysis,” in Proceedings: Asian Conference on Computer Vision. pp.188-192, 2002.

[19] L. Gu, S. Z. Li and H. J. Zhang, “Learning Probabilistic Distribution Model for Multiview Face Detection,” in Proceedings: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01). Vol. 2, 2001.

[20] Z. Liposcak and S. Loncaric “A scale-space approach to face recognition from profile,” International conference on Computer Analysis of Images and Patterns. Ljubljana, Slovenia, September 1-3, 1999.

[21] A. Pentland, B. Moghaddam and T. Starner, “View-based and modular eigenspaces for face recognition,” in Proceedings: IEEE conference on computer vision and pattern recognition, 1994.

[22] P. Penev and J. Atick, “Local feature analysis: a general statistical theory for object representation,” *Network: Computation Neural Syst*. Vol. 7, p. 477, 1996.

[23] A. Lanitis, C. J. Taylor and T. F. Cootes, “Automatic face identification system using flexible appearance models,” *Image Vis. Comput*. Vol. 13, pp. 393-401, 1995.

[24] H. Ling and D. W. Jacobs, “Deformation Invariant Image Matching,” IEEE International Conference on Computer Vision, Vol. 2, pp. 1466-1473, 2005.

[25] A. P. Witkin, “Scale-space filtering,” in Proceedings: 8th. International Joint Conference on A.I. pp 1019-1022, 1983.

[26] C. Sanderson and K. K. Paliwal. “Polynomial Features for Robust Face Authentication,” in Proceedings: IEEE International Conference on Image Processing (ICIP), Vol. 3, pp. 997-1000, 2002.

[27] P. J. Grother, R. J. Micheals and P. J. Phillips, “Face Recognition Vendor Test 2002 Performance Metrics,” in Proceedings: 4th International Conference on Audio Visual Based Person Authentication, 2003.

[28] R. Beveridge, D. Bolme, M. Teixeira and B. Draper, “The CSU Face Identification Evaluation System User’s Guide,” Computer Science Department Colorado State University. May 1, 2003.

[29] B. Horn and B. Schunck, “Determining Optical Flow,” *Artificial Intelligence*. Vol. 17, No. 1-3, pp. 185-203, 1981.

[30] I. Cohen, N Sebe, A Garg, L Chen and T Huang, “Facial Expression Recognition from Video Sequences: Temporal and Static Modeling,” *Computer Vision and Image Understanding*. Vol. 91, pp 160-187, 2003.